No. 2015-1652, -1653

United States Court of Appeals for the Federal Circuit

SOFTWARE RIGHTS ARCHIVE, LLC,
Appellant,

v.

FACEBOOK, INC., LINKEDIN CORPORATION, TWITTER, INC.,
Cross-Appellants.

Appeals from the United States Patent and Trademark Office, Patent Trial and Appeal Board in No. IPR2013-00481.

APPELLANT OPENING BRIEF

Martin M. Zoltick
Rothwell, Figg, Ernst & Manbeck, P.C.
607 14th Street, N.W., Suite 800
Washington, DC 20005
Phone: 202-783-6040

Victor Hardy
Minghui Yang
DiNovo Price Ellwanger & Hardy LLP
7000 North MoPac Expy, Suite 350
Austin, TX 78731
Phone: 512-539-2626

July 27, 2015

Attorneys for Appellant

CERTIFICATE OF INTEREST

I, Minghui Yang, counsel for Appellant Software Rights Archive, LLC, certify the following:

1. The full name of the party represented by me is Software Rights Archive, LLC.

2. Software Rights Archive, LLC is the real party in interest.

3. SRA, LLC is the parent company of Software Rights Archive, LLC.

4. All parent corporations and publicly held companies that own 10 percent or more of the stock of this party are: None

5. The law firms and the partners and associates that appeared for this party in the *Inter Partes* Review case at the U.S. Patent and Trademark Office, Patent Trial and Appeal Board, or are expected to appear in this Court are:

ROTHWELL, FIGG, ERNST & MANBECK, P.C.
Martin M. Zoltick
Nancy J. Linck
Soumya P. Panda
607 14th Street, N.W., Suite 800
Washington, DC 20005
Phone: 202-783-6040

DINOVO, PRICE, ELLWANGER & HARDY, LLP
Victor G. Hardy
Minghui Yang
7000 North MoPac Expy, Suite 350
Austin, TX 78731
Phone: 512-539-2626

Date: July 27, 2015

Respectfully submitted,

/s/Minghui Yang
Minghui Yang
DiNovo Price Ellwanger & Hardy LLP
7000 North MoPac Expressway
Suite 350
Austin, Texas 78731
Telephone: 512-539-2626
Facsimile: 512-539-2627
E-mail: myang@dpelaw.com

TABLE OF CONTENTS

TABLE OF ADDENDA

TABLE OF AUTHORITIES

Cases

Statutes

Regulations

TABLE OF ABBREVIATIONS

Parties

Petitioners	Facebook, Inc., LinkedIn Corp., and Twitter Inc.
SRA	Software Rights Archive, LLC

Prior Art

Fox Papers	Fox Thesis, Fox SMART, Fox Collection
Fox Thesis	Edward A. Fox, Extending the Boolean and Vector Space Models of Information Retrieval with P-Norm Queries and Multiple Concept Types, (Aug.1983) (Ph.D. dissertation, Cornell Univ. Dept. of Comp. Sci.)
Fox SMART	Edward A. Fox, Some Considerations for Implementing the SMART Information Retrieval System under UNIX, (Sept. 1983) (Ph.D. dissertation, Cornell Univ. Dept. of Comp. Sci.)
Fox Collection	Edward A. Fox, Characterization of Two New Experimental Collections in Computer and Information Science Containing Textual and Bibliographic Concepts, (Sept. 1983) (Ph.D. dissertation, Cornell Univ. Dept. of Comp. Sci.)
Fox Envision	Edward A. Fox, et al., Users, User Interfaces, and Objects: Envision, a Digital Library, 44 J. Am. Soc. Inf. Sci., no. 8 at 480-91 (Sept. 1993)

Fox 1991	Edward A. Fox, et al., Integrating Search and Retrieval with Hyptertext, In Hypertext/Hypermedia Handbook, ed. E. Berk and J. Devlin, McGraw-Hill, New York, 1991, 329-355

Terms

’571 Patent	U.S. Pat. No. 6,233,571
Reviewed Claims	Claims 12, 21, and 22 of the’571 Patent
Board	Patent Trials and Appeal Board
CACM	Communications of the Association of Computing Machinery
did	manually assigned document identifier
IPR	*Inter Partes* Review
IR	Information Retrieval
ISI	Institute for Scientific Information
au	author
bc	bibliographic coupling
cc	co-citation
cd	co-citation direct
cr	*Computer Review* categorization
ln	direct links
tm	terms

STATEMENT OF RELATED CASES

Pursuant to Rule 47.5 of the Federal Circuit Rules, SRA states that no other appeal in or from the *Inter Partes* Review proceeding, No. IPR2013-00481, before the Patent Trial and Appeal Board of the U.S. Patent and Trademark Office (the "Board") has been before this or any other appellate court. This appeal is a companion case to Appeal No. 15-1648 and Consolidated Cross-Appeal Nos. 15-1649, 1650. The three appeals have been assigned to the same merits panel pursuant to D.I. 2.

I. STATEMENT OF JURISDICTION

This appeal is from an IPR for the ’571 Patent instituted by the Board under the American Invents Act (“AIA”). On July 27, 2012, SRA filed its complaint in the United States District Court for the Northern District of California against Petitioners, asserting infringement of the ’571 Patent, among other patents. On July 30, 2013, the Petitioners filed a Petition for *inter partes* review of claims 12, 21, and 22 of the ’571 Patent. JA01000. On November 5, 2013, SRA filed its Patent Owner’s Preliminary Response to the Petition. JA01125. On February 3, 2014, the Board instituted an IPR as to claims 12, 21, and 22 (the “Reviewed Claims”) on the grounds of obviousness in view of the Fox Papers. JA01158.

The Board issued its Final Written Decision (“decision”) on January 29, 2015. JA00001. In its decision, the Board found claim 21 patentable but found claims 12 and 22 as obvious based on the Fox Papers. JA00015; JA00017; JA00029.

SRA timely filed its Notice of Appeal on April 1, 2015. JA01486. This Court has jurisdiction over this appeal pursuant to 28 U.S.C. § 1295(a)(4)(A), 35 U.S.C. § 319, and 35 U.S.C. § 141(c).

II. STATEMENT OF THE ISSUES

In finding the Reviewed claims obvious, whether the Board erred by:

1. failing to provide a persuasive “explanation” as to how the combination of

the references discloses identifying and analyzing hyperlinks?

2. failing to provide a persuasive "explanation" as to why one skilled in the art would use indirect hyperlink relationships to search in view of the empirical evidence that demonstrate indirect relationships as claimed harm search results?
3. failing to consider the prior art as a whole and important objective indicia of nonobviousness?
4. failing to properly arrive at the ultimate conclusion of nonobviousness?
5. finding that PageRank did not have a nexus with the claimed invention?

III. STATEMENT OF THE CASE

A. Preliminary Statement

In the Board's final written decision, the Board correctly concluded that claim 21 is patentable over Fox Thesis, Fox Smart, and Fox Envision (collectively "Fox Papers"). JA00015. However, the Board erroneously invalidated claims 12 and 21 over these references. JA00017; JA00029.

Claims 12 and 22 of the '571 Patent require (1) identification of hyperjump data or URLs; (2) analyzing hyperjump data or URLs for indirect relationships; and (3) using URLs to display search results. JA05083(52:38-56); JA05090-JA05091(2:57-3:12). None of these three steps pertaining to URLs are taught or suggested by the Fox Papers. Indeed the word "universal resource locator," does

not even appear in Fox Envision or any other of the Fox papers relied upon by the Board. JA01260-JA01266.

Most significantly however, the analysis of hyperlinks on the web to improve information retrieval, as required by the ’571 Patent claims, was consider a revolutionary technological breakthrough in 1998. JA12377-JA12378; JA12482-JA12491; JA12472-JA12473; JA12477-JA12478. Prior to that, the art had failed to appreciate that hyperlinks contained “meaningful” information that could improve search:

> We originally developed page rank[] kind of playing around with studying all the links on the web. And that too was a pretty ***<u>revolutionary idea</u>*** though it seems very simple that you could even just collect [the links of the web and] ***do <u>anything</u> meaningful*** ….
> I really credit Larry pursuing ***that idea that it’s even worth collecting the graph.*** And then that you could ***run <u>any</u> kind of processing*** on it.

JA12377-JA12378.

The notion of further extending the analysis from bibliographic citations to hyperlink references of computer databases such as the World Wide Web is completely absent from these papers and would be considered a major breakthrough in technology by the industry nearly 15 years after the Fox papers. JA12377-JA12378; JA12482-JA12491; JA12472-JA12473; JA12477-JA12478.

Furthermore, claims 12 and 22 of the ’571 Patent require an analysis of hyperlinks for <u>indirect relationships</u>, which the Board erroneously found obvious. The United States Supreme Court in *KSR Int’l Co. v. Telefex Inc.*, 550 U.S. 398,

401 (2007) explains that an obvious combination is one that seeks to take (1) "familiar elements" and use them in (2) "known methods" to (3) "yield predictable results." Here, the prior art as a whole demonstrates that the Fox experiments on indirect citations involved the use of (1) only experimental elements that (2) yielded highly unpredictable and harmful results (3) in a method that is empirically shown not to work for the intended purpose of computerized search. JA12379-JA12392; JA15042-JA15046; JA12398-JA12399(¶171).

The Board found the claims obvious based upon experiments performed on printed documents in the early 1980s to determine the utility of citation relationships and other factors for searching, even though the results of those experiments overwhelmingly demonstrated that the claimed use of indirect relationships was both too harmful and too unreliable for use in computerized search. Indeed, with respect to "*cc*" and "*bc*" (the only indirect relationships relied upon by Petitioners), Dr. Fox himself concludes that "*cc* is not really needed and *bc* is probably not either." JA12368.

These widely published studies from one of the most prominent information retrieval ("IR") research departments in the world did not make obvious the use of citation relationships to enhance research. Rather, for decades these experiments and other publications discouraged the IR field from using non-semantical relationships (i.e, referential relationships) for search of any type. JA12362-

JA12379. Indeed, three years after the Fox Papers (*i.e.*, the prior art that allegedly rendered the use of citation relationships obvious), Gerald Salton, Fox's thesis advisor, stated:

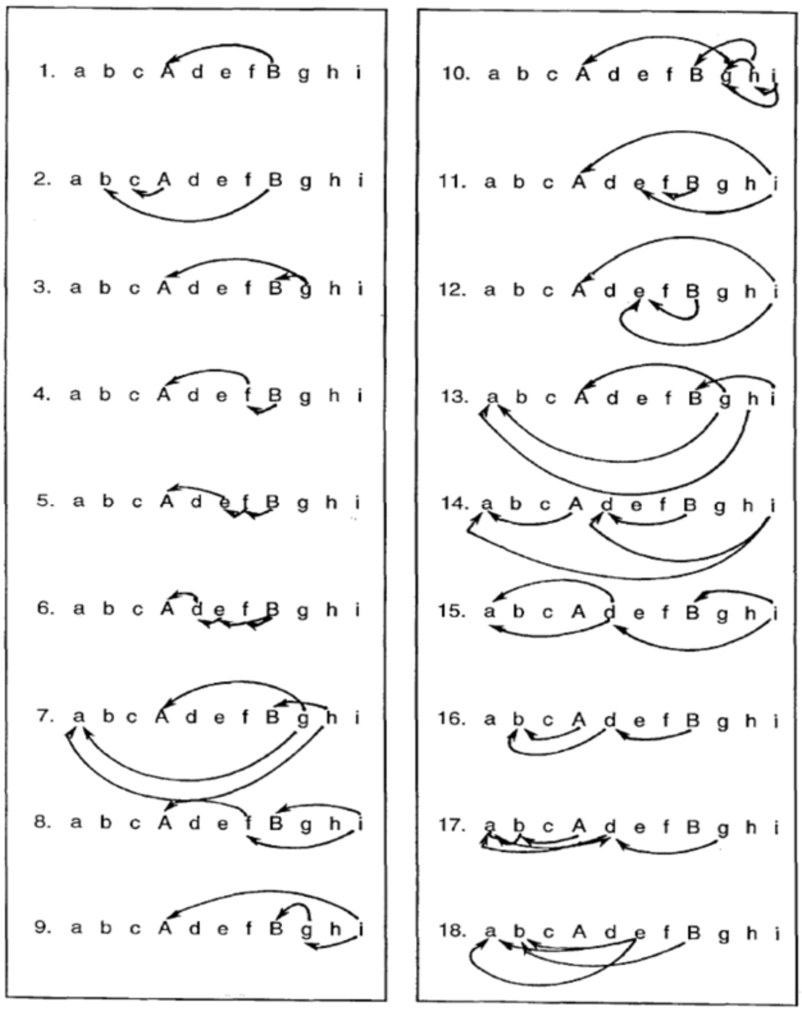
> Since no obvious way exists for distinguishing the positive from the negative effects, the citation methodology cannot be recommended for inclusion in practical retrieval environments.

JA12398. Consequently, the early search engines relied exclusively on semantical relationships (*i.e.*, relationships based upon the meaning of words). There was a complete absence of any use of indirect relationships for nearly 15 years in a non-experimental search system after these experiments in the early 1980s. JA12373.

B. Factual Statement

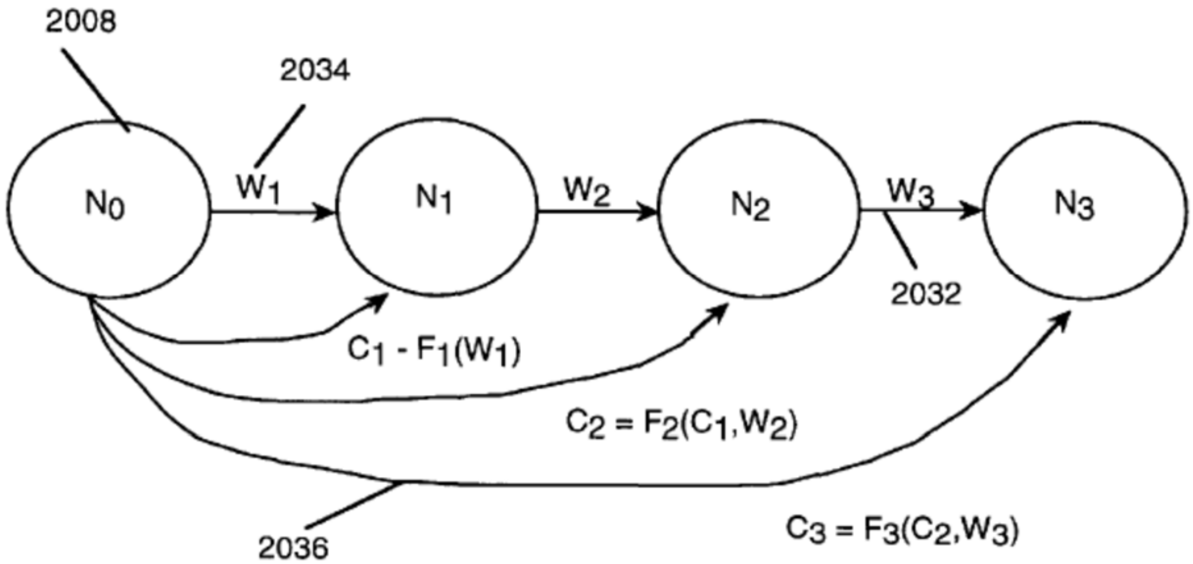
The '571 Patent relates to the use of direct and indirect citations to search for objects in a database. Since these relationships are based upon a referential nature of a citation, rather than the occurrence or meaning of words, they are non-semantical. The terms "direct" and "indirect" refer to the type of reference of a citation relationship. As construed by the District Court and the Board, a direct relationship is when "one object directly cites or refers to another object." *Facebook, Inc., et al. v. Software Rights Archive, LLC*, IPR2013-00478, Paper 58 at *8 (PTAB 2015). For example, A&B are directly related when A contains a citation that refers to B. An indirect relationship refers to two objects being connected by a chain of citation references, and therefore being "indirectly"

referenced. *Id*. The Patent identifies 18 patterns of empirically referential relationships that may be embodied into a numerical representation that is used for searching:



JA05028; JA05064(14:22-56). The patent specification further discloses a unique analysis of indirect relationships called the cluster link generator. By assigning weights to each direct relationship that comprises a potential indirect relationship and then aggregating those weights to determine a weight of a path, the cluster link

generator can analyze any type of indirect relationship within a specified number of links:



Another unique feature of a "cluster link" is that it analyzes multiple indirect relationships of multiple different types to generate a single value for searching. JA05009; JA05068(22:6-40).

Another novel feature disclosed in the specification is Egger's recognition that hyperlink and other non-bibliographic citation relationships of computer databases should be analyzed. JA05081-JA05082(48:63-49:36). Analysis of these types of relationships from computer database would revolutionize the search industry. Importantly, the system describes an inventive step where hyperlink references are analyzed as citations:

> In another embodiment, the Proximity Indexing Application Program (Program) 62 organizes and categorizes the crawled links 2004 using

> the statistical techniques and empirically generated algorithms described earlier in this application. **The Program 62 treats URL addresses as citations and web pages as textual objects.**

JA05082(50:4-9). As discussed *infra* at 25-29, use of this relationship was considered a major breakthrough in technology in the late 1990s.

Claim 12 recites:

> A method for visually displaying data related to a web having identifiable web pages and Universal Resource Locators with pointers, comprising:
>
> choosing an identifiable web page;
>
> identifying Universal Resource Locators for the web pages, wherein the identified Universal Resource Locators either point to or point away from the chosen web page;
>
> *analyzing Universal Resource Locators*, including the identified Universal Resource Locators, wherein Universal Resource Locators which have an indirect relationship to the chosen web page are located, wherein the step of analyzing further comprises *cluster analyzing the Universal Resource Locators for indirect relationships*; and
>
> displaying identities of web pages, *wherein the located Universal Resource Locators are used to identify web pages.*

JA05083(52:38-56). As can be seen above, claim 12 (and claim 21) requires an analysis "for indirect relationships" for purposes of search. Fox's experimental results concerning *bc* and *cc*, the only analyzed indirect relationships of Fox, overwhelmingly demonstrates that the use of these relationships harm search results. *Infra* at 11-17. Claim 22 is nonobvious for substantially the same reasons

provided for claim 12 based on the features of analyzing hyperlink relationships and indirect relationships recited in claim 22.

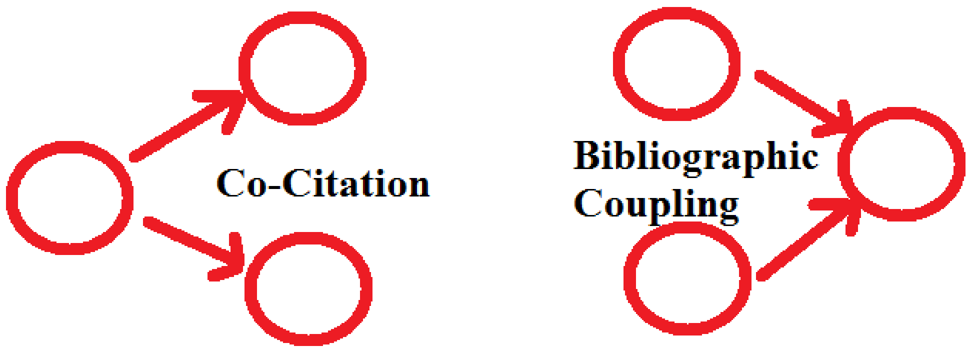
1. The Fox Papers

The Fox Papers published in 1983 comprise Fox Thesis, Fox SMART, and Fox Collection. The Petitioners rely upon the teaching of the Fox Papers describing experiments with two collections: CACM and ISI. These experiments were methods applied, by hand, to printed bibliographies and were not performed on any computer database of objects with relationships, as claimed. JA12358-JA12359; JA12369.

Fox SMART describes the SMART computer system created by Gerald Salton used to test information retrieval methods. JA05908. Fox Thesis reported on the experimental results of SMART system on the available collections. JA05773-JA05800.

The experiments used a number of data types including: terms [in titles of documents only] (*t*), *Computer Review* subject matter categories (*cr*) ; co-citations (*cc*); co-citation direct (*cd*); direct citations (*ln*); authors (*au*) and bibliographic couplings (*bc*). JA05723. The *ln* subvector represents direct relationships between two articles that cite each other. The *bc* subvectors represents two articles that cite a common article, and the *cc* subvectors represents a frequency with which two

documents are cited together by other documents. *Id*. They relied upon by the Petitioners as the recited "indirect relationships." JA01019-JA01020.



2. The Fox Papers, When Fairly Considered as a Whole, Teach that Using BC and CC Harm Search Results

Petitioners rely upon tests performed on the *cc* and *bc* subvectors in SMART as the claimed "cluster analysis for indirect relationships." JA01019-JA01020. However, one would not have used *bc* and *cc* in the claimed arrangement to analyze hyperlinks for indirect relationships because the experimental results of the Fox Papers demonstrate that such use is far more likely to harm search results than to improve them. JA12379-JA12392; JA15042-JA15046; JA12398-JA12399(¶171).

Chapter 8 of the Fox Thesis details nine experiments using *bc*, *cc* and other subvectors to determine whether these subvectors would lead to any improvement in search. JA15044-JA15045; JA12386-JA12387. Many of Fox's tests are directed towards combined subvectors for which *bc* and *cc* are just two

of many data types included within a single subvector. The Fox experiments are directed to the use of subvectors generally, with no particular focus on indirect relationship subvectors, thus many of Fox's tables do not report directly on the specific impact that *bc* and *cc* may have on the results. JA12386-JA12387. Since the issue here is whether it would be obvious to use the claimed indirect relationships, Dr. Jacobs analyzed the data of these tables to determine the specific impact that can be attributable to *bc* and *cc* (as opposed to other data types in the combined vectors) as shown through the Fox experiments.[1] By using terms and direct links as a comparison, the *specific* contribution of indirect relationships on search results can be determined. Whereas the combined subvector including *bc* or *cc* may at times show an improvement, the specific contribution to that improvement of the *bc* and *cc* data types of the combined vector is almost always negative. Neither Petitioner nor the Board has challenged the accuracy of Dr. Jacobs's analysis on the specific impact that *bc* and *cc* had on search results. Dr. Jacobs's analysis of all 9 experiments are set forth below:

---

[1] By examining the results of each table, the specific impact of a given variable can be isolated. For example, if "*tm+ln+bc*" results in a precision of .3187 which is a 1.1% improvement over terms and if "*tm+ln*" by itself results in a precision of .3431 which is a 8.8% improvement, the incremental improvement of *bc* by itself can be calculated by subtracting the former average precision by the latter and then dividing by the latter again (approximately (1.1%-8.8%) or -7.7%). JA15045-JA15046; JA05598.

Experiment	Result	Experiment	Result
8.2-ISI (*cc*)	**-37% (vs terms)**	8.11-CACM (*bc*)	**-6.3% (vs terms)**
8.2-ISI (*cd*)	**-54% (vs terms)**	8.11-CACM (*cc*)	**-1.4% (vs terms)**
8.3-ISI (*cc*+terms)	**-6.2% (vs terms)**	8.11-CACM (*bc*+*cc*)	**-2.9% (vs terms)**
8.3-ISI (*cc*+other subvectors)	**-7.8% (vs terms)**	8.11-CACM (*bc*)	**-7.4% (vs *ln*)**
8.3-ISI (*cc*+terms emphasized)	+4.5 to 6.1% (vs terms)	8.11-CACM (*cc*)	**-6% (vs *ln*)**
8.7-ISI (*cc*+terms regression)	+5 to 6% (vs terms)	8.11-CACM (*bc*+*cc*)	**-4.1% (vs *ln*)**
8.8-CACM (*bc*)	**-62% (vs terms)**	8.12-CACM (*bc*)	**-26.5% (vs terms)**
8.8-CACM (*cc*)	**-51% (vs terms)**	8.12-CACM (cc)	**-29% (vs terms)**

Experiment	Result
8.13-CACM (*bc*)	**-11.1%** (vs *ln*)
8.13-CACM (*cc*)	**-3.7%** (vs terms)
8.13-CACM (*bc*+*cc*)	**-11.5%** (vs *ln*)
8.13-CACM (all subvectors regression)	undetermined but less than +2.1% divided among *au*, *cr*, *bc*, and *cc* (no positive contribution from indirect relationships because *bc* and *cc* weighed at less than 1% and 0% respectively)

JA12594-JA12595; JA12380-JA12385. When the specific contribution of *bc* and *cc* is isolated, 7 out of 9 of the experiments show *only* deterioration from their use. JA15044-JA15045; JA12386-JA12387. The remaining experiments either show a neutral impact or an insignificant improvement insufficient to establish a connection between these relationships and search improvement. *Id*.

Although Petitioners seize on a few of the results that showed some slight improvement (ISI in Table 8.3;8.7), the table above clearly shows when all the

teachings of the Fox Papers are fairly considered, they overwhelming demonstrate that use of *bc* and *cc* harms search results for the vast majority of queries. A skilled artisan would thus understand that the use of *bc* and *cc* is far more likely to hurt search results than to help and would not be motivated to use Fox's methods regarding *bc* and *cc* on any collection, much less the claimed computer database. *Id*.

Dr. Salton is considered the "father of information retrieval," created the vector space model and SMART system used by Fox, and was the Thesis Advisor on the Fox Papers. JA12363. He also conducted tests using bibliographic subvectors on the same CACM and ISI collections as Fox. Dr. Salton stated that a 10% improvement is the "normal" threshold for determining significant results. JA12389. Importantly, <u>none</u> of Fox's test results come close to Salton's 10% improvement threshold. JA15044-JA15045; JA12386-JA12387.

Furthermore, there "must be evidence that 'a skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would select the elements from the cited prior art references for combination ***in the manner claimed***.'" *Ecolochem, Inc. v. S. Cal. Edison Co.*, 227 F.3d 1361, 1375 (Fed. Cir. 2000); *see also In re Kotzab*, 217 F.3d 1365, 1371 (Fed. Cir. 2000). Claim 12 requires the identification of direct links for objects in the database:

> identifying Universal Resource Locators for the web pages, wherein the identified Universal Resource *Locators either point to or point away from the chosen web page;*

JA05083(52:38-56). After the direct relationships are identified, the claimed method then requires the use of a cluster analysis of indirect relationships for searching:

> *analyzing Universal Resource Locators*, including the identified Universal Resource Locators, wherein Universal Resource Locators which have an indirect relationship to the chosen web page are located, wherein the step of analyzing further comprises *cluster analyzing the Universal Resource Locators for indirect relationships*; and
>
> displaying identities of web pages, *wherein the located Universal Resource Locators are used to identify web pages.*

Id. Thus, the claimed method specifically requires one to use indirect relationships for searching when one has available the direct relationships for use in searching. JA12368-JA12369.

Petitioners rely upon the use of *bc* and cc as the only analyzed indirect relationships in the Fox papers. JA01019-JA01020. Fox's experiments unequivocally demonstrate that, after identifying the direct relationships, a skilled artisan would have no motivation to perform the additional steps pertaining to cluster analyzing "for indirect relationships" for search. These additional steps concerning indirect relationships would not only involve an extreme multi-year effort using Fox's hand methods, but would actually result in harming the search results. JA150065-JA150066; JA12368-JA12369; JA12380-JA12385; JA15047-

JA15048; JA15057-JA15059. Although there are two cases that show a slight improvement of indirect relationships over terms only, **there is no case** where the Fox Papers show the use of *bc* and *cc* improves search results when direct links (*ln*) are present, as specifically required by the claims. *Id*. In **every single test** of Fox that addresses this matter, the use of *bc* and *cc* (in combination with direct links or alone or with each other) has resulted in deterioration of search results:

Experiment	Result	Experiment	Result
8.8-CACM (*bc*)	**-43.5%** (vs *ln*)	8.11-CACM (*bc*+*cc*)	**-4.1%** (vs *ln*)
8.8-CACM (*cc*)	**-26.7%** (vs *ln*)	8.12-CACM (*bc*)	**-30%** (vs *ln*)
8.11-CACM (*bc*)	**-7.4%** (vs *ln*)	8.12-CACM (cc)	**-32.4%** (vs *ln*)
8.11-CACM (*cc*)	**-6%** (vs *ln*)		

JA12644; JA12380-JA12385. Accordingly, there is no teaching in the Fox Papers that would have suggested that *bc* or *cc* would improve results in the context that the claims specifically require and they could not suggest the methods of the Reviewed Claims. *Id*. Fox himself reaches this same conclusion and expressly states that the *cc* and *bc* subvectors should **not** be used if one has other subvectors (which includes the direct links subvector "*ln*"):

> [I]t can be inferred, however, that with the other subvectors present *cc* is not really needed and *bc* is probably not either.

JA12368. Notably, Fox also explicitly excludes *bc* and *cc* from his proposed "recipe" for the subvectors, and instead teaches using just the direct links subvector

as the only non-semantical subvector because of "effectiveness tests":

> ***The recipe proposed is to at least employ terms (tm), some manually assigned categorization scheme (cr), and direct links between documents (ln).*** When bibliographic information is only available among articles in a collection the simplest form of that information, references (*ln*) [*i.e.*, the direct links vector], seems to be the most reliable and most useful of all the types considered (bc, ln, cc). ***The ln subvectors*** are typically longer than the other two and are easier to obtain ***so use of them is encouraged by*** practicality considerations as well as ***effectiveness tests***.

JA12365. In short, when all of the teachings of the Fox Papers including experimental results are fairly considered, it is clear that they teach that an analysis of indirect relationships using the Fox method is highly likely to harm search results—and always harmful within the claimed arrangement which requires one to identify direct relationships that could be used for searching.

3. Fox's Methods Using Indirect Relationships Suffer from a Reliability Problem that Makes Them Unfit for their Intended Purpose

The Fox experiments also demonstrate that the use of citation relationships for search suffered from a reliability problem that made them unfit for computerized search. The impact of the indirect subvectors on search performance across retrieval environments was unpredictable, ranging from 5-6%, a slight improvement, to -62%, a very significant degradation in search performance. JA12380-JA12385; JA15057-JA15059. For any given collection, indirect relationships were far more likely to result in degradation of search results than to

help them and, consequently, the citation methodology was unsuited for use in automated retrieval. JA12397-JA12398.

It is hardly obvious to deploy methods far more likely to hurt search results than to help them, especially given the extensive, two year effort required to index a mere 6,000 articles using Fox's hand methods. JA01283. In 1986, Dr. Salton addresses how a skilled artisan would interpret data revealing a strong potential for degradation of search results. Dr. Salton sought to test the performance of using bibliographic citations for search across different "retrieval environments" (different queries and documents) to determine whether they could be reliably used in an operational search system. JA10859. His experiments were conducted using bibliographic subvectors on the same CACM and ISI collections used in the Fox Papers.

In his experiments, Dr. Salton achieves a significant "**30 percent**" **improvement** in precision from direct links in the CACM collection:

> The CACM results of [Table 3] show clearly that the addition to the document terms of title words from bibliographically related documents is beneficial, since the retrieval effectiveness improves by *about 30%* on average for the citing+cited method….

JA10862. However, when trying to apply the same method to the ISI collection, he experienced a **1-7 percent deterioration**:

> Unfortunately, this optimistic conclusion is not maintainable when the CISI results of Table 4 are considered. In that case, ***none*** of the

> bibliographic expansion method ***proved beneficial*** for the more effective tf x idF weighting system***, the deterioration in effectiveness ranging from about 1 percent to as much as 7 percent for the citing+cited method***.

JA10862-JA10863. In other words, the same methods that seemed to be effective on the CACM queries actually degraded the results when applied to different queries of the ISI collection. *Id.* When Dr. Salton is confronted with a method that sometimes results in positive results (+30 percent) and sometimes results in negative results (-7 percent deterioration), he ultimately concludes that the method in unsuited for its intended purpose of computerized search:

> ***Since no obvious way exists for distinguishing the positive from the negative effects,*** the citation methodology cannot be recommended for inclusion in practical retrieval environments.
>
> ***
>
> Overall, the procedure is not sufficiently reliable to warrant incorporation into operational automatic retrieval systems.

Id.

Salton's conclusion is direct and compelling evidence of what a skilled artisan would conclude in face of the evidence of unpredictable, disparate experimental results of the Fox experiments that reveal *bc* and *cc* are likely to harm search results. JA12396-JA12397. Unlike Petitioners, Salton did not seize one positive result and ignore all the negative results; rather he looked at the range of outcomes and concluded that the citation methodology was too unreliable for use in a system because of the strong potential for harming search results. This same problem of

extreme variation and potential harm to search results that led Dr. Salton to this conclusion was also apparent in Dr. Fox's 1983 experimental work. JA12396-JA12398; JA15057-JA15059. Indeed, the data in the Fox Papers suggests a far worse reliability problem than that in Salton's tests. One skilled in the art would see the following test results with respect to the CACM collection for *bc* and *cc*:

Experiment	Result	Experiment	Result
8.8-CACM (*bc*)	**-62%** (vs terms)	8.12-CACM (*bc*)	**-26.5%** (vs terms)
8.8-CACM (*cc*)	**-51%** (vs terms)	8.12-CACM (cc)	**-29%** (vs terms)
8.11-CACM (*bc*)	**-6.3%** (vs terms)	8.13-CACM (*bc*)	**-11.1%** (vs *ln*)
8.11-CACM (*cc*)	**-1.4%** (vs terms)	8.13-CACM (*cc*)	**-3.7%** (vs terms)
8.11-CACM (*bc*+*cc*)	**-2.9%** (vs terms)	8.13-CACM (*bc*+*cc*)	**-11.5%** (vs *ln*)
8.11-CACM (*bc*)	**-7.4%** (vs *ln*)	8.13-CACM (all subvectors regression)	undetermined but less than +2.1% divided among *au*, *cr*, *bc*, and *cc* (no positive contribution from indirect relationships because *bc* and *cc* weighed at less than 1% and 0% respectively)
8.11-CACM (*cc*)	**-6%** (vs *ln*)		
8.11-CACM (*bc*+*cc*)	**-4.1%** (vs *ln*)		

JA12594-JA12595; JA12380-JA12385. As reflected above, the impact of *bc* and *cc* for the CACM range from -62% degradation to potentially +2.1% improvement (but likely to be "0"[2]).

---

[2] JA12388.

Experiment	Result
8.2-ISI (*cc*)	**-37%** (vs terms)
8.2-ISI (*cd*)	**-54%** (vs terms)
8.3-ISI (*cc*+terms)	**-6.2%** (vs terms)
8.3-ISI (*cc*+other subvectors)	**-7.8%** (vs terms)
8.3-ISI (*cc*+terms)	+4.5 to 6.1% (vs terms)
8.7-ISI (*cc*+terms regression)	+5 to 6% (vs terms)

JA12647; JA12380-JA12385. With respect to the ISI, *bc* and *cc* ranged from negative -54% deterioration to potentially +6 % percent improvement. *Id*. Thus, Fox's CACM and ISI results actually show far more extreme degradation (-54% and -62% (Fox) vs. -1% to -7% (Salton)) and far less beneficial results (+6% (Fox) v. +30% (Salton)) than the variation of results that led Dr. Salton, the then most prominent researcher in the world, to conclude that the citation methodology was unfit for its intended purpose. JA12398; JA15057-JA15059. Likewise, a skilled artisan would also view Fox's results as indicating that Fox's method pertaining to *bc* and *cc* suffered from a serious reliability problem that made it unfit for use in search. Consequently, the skilled artisan would not be motivated to use this method in any search system, much less one for the claimed database relationships.

4. The Art of Record as a Whole Confirms that It was not Obvious to use Indirect Relationships for Search

The negative experimental results of Fox must be viewed in the larger context of the art of record. The Petitioners' apparent belief that the successful use

of indirect relationships in a search system was a "predictable result," or obvious to one of ordinary skill, is factually unsupported. JA12362-JA12379. To the contrary, despite nearly 40 years of testing citations, the prior art as a whole indicates a field that struggled with finding any empirically supported use of indirect relationships for search until after the Patent. *Id*. Accordingly, a skilled artisan would be skeptical of any method that used citation relationships and would demand strong empirical evidence supporting its utility for search—a feature lacking in the experimental results of the Fox method concerning *bc* and *cc*.

Dr. Jacobs sets forth a timeline of the work done on citation relationships. JA12362-JA12379. The little work that had been done prior to the '571 Patent consisted of a few isolated experiments using small-scale, prepared data (such as the CACM collection). *Id*. This work is replete with negative or inconclusive experiments and explicit skepticism by the leading experts. The utility of citation relationships for search remained an open question studied by leading research experts for nearly 15 years after the work of Fox that the Petitioners allege rendered the use of indirect relationships obvious to persons of ordinary skill. *Id*.

Dr. Jacobs's timeline included experiments from Salton, Fox's thesis advisor and creator of the vector space model used by Fox. JA12397-JA12401. Salton specifically noted that the "initial reaction must clearly be one of

skepticism" toward the use of bibliographic citation vectors and that "[b]ecause of these and other variations, citation and reference lists have not generally been used as an indication of document content." JA12401. On pages 445-46, Salton lists many potential reasons why citations may be an unreliable indicator of content including self-selection, survey article problem, and availability of relevant works; thereby demonstrating that their utility is not "predictable" or "obvious" to ordinary practitioners. *Id.* Salton ultimately concludes that his experiments fail to establish the utility of using citations for search:

> ***Clearly no proof*** has been presented in this study that citations do in fact play a significant role in automatic document retrieval systems.

JA12364. In 1986, after conducting additional experiments with citation relationships, Dr. Salton stated three years *after* the Fox thesis:

> Since no obvious way exists for distinguishing the positive from the negative effects, the citation methodology cannot be recommended for inclusion in practical retrieval environments.
> ***
>
> ***Overall, the procedure is not sufficiently reliable to warrant incorporation into operational automatic retrieval systems***.

JA12398. Others in the field made similar comments about the use of bibliographic relations. JA12375.

In 1987, four years after the Fox experiments, Fox's graduate student Gary Nunn further analyzed the original 1983 Fox data. JA12399-JA12400. Nunn attempted to determine if Fox's method could be extended to other collections.

They could not. Nunn described his results as "disappointing" because the results obtained on one half of the CACM and ISI collections could not generalize to the other half of the same collection. These results demonstrated that the isolated cases where citation links showed improvement to search could <u>not</u> be replicated on other queries made on a different part of the <u>same</u> collection. JA12399. Ultimately, Nunn concludes that Fox's methods were not generalizable to other collections and suggests further research is needed to find coefficients that could work on other collections:

> The author of this study would like to suggest that further research might be pursued along two paths…. The purpose of this … [would be] to try to develop coefficients that *could be* generalized to other collections.

JA12399-JA12400; JA10917.

In 1992, Frei and Steiger continued to discourage the use of citation relationships: "Retrieval experiments in a collection of bibliographic references showed that following citations – a kind of referential link—***produces ambiguous results*** ….***The hope is that our semantic links*** contain the information necessary…" JA12375. That same year, Ledwith also expressed skepticism at the reliability of Fox's 1983 methods:

> [D]espite the significant efforts to explore and develop these models, ***there remain concerns about the models' utility for the searching of large scientific databases***. Using the p-norm retrieval experiment described in ***<u>Fox (1983)</u>*** as an example, I will present my three major concerns ***… the <u>reliability</u> of extrapolating the performance of***

> ***research systems that use the collection to a system to search a file over 750 times larger than the collection is <u>highly questionable</u>***… JA12375-JA12376.

The negative experimental results and skepticism by experts towards the use of citation relationships for years after Fox is objective evidence that the Fox Papers did not render obvious the use of indirect relationships for computerized search. To this day, no empirical test has confirmed that *bc* and *cc alone* (without other higher order citation relationships like Egger) improves search results in a reliable fashion. Petitioners made no attempt in their papers to address these teachings, nor did the Board in its decision comment substantively on the unreliability evidence.

5. The Objective Evidence Confirms that the Fox Methods Concerning BC and CC did Not Render Searching with Indirect Relationships Obvious

In 1963, Salton first experimented with indirect citation relationships. JA12363. Fox's experiments occurred in 1983. JA12365. In the 35 years since these relationships were initially studied (and 15 years since the Fox experiments allegedly rendered obvious using indirect relationships), no non-experimental system used indirect relationships for search prior to 1998 (except the system of Egger). JA12373; JA12461-JA12462. Indeed, since the SMART system was only used to analyze relationships in paper copies, no system at all was used to analyze indirect relationships of a computer databases as claimed prior to the '571 Patent.

Id.

The complete absence of any use of the claimed method for 35 years from Salton (and 15 years from Fox) is compelling evidence that the Fox's experiments did not render the use of indirect relationships obvious. JA12396-JA12363. This absence also strongly corroborates Dr. Jacobs's testimony that the skilled artisan would understand from Fox's results (and others) that indirect relationships *bc* and *cc* actually harm search results. JA12363. Neither the Petitioners in their papers nor the Board in its decision address this compelling fact or offer any reason to explain this absence.

Furthermore, the inference of nonobviousness from this absence of use is even stronger here, where the analysis of the claimed non-semantic computer database relationships was considered a major breakthrough nearly 15 years after the Fox Papers allegedly rendered their use obvious. JA12472-JA12473; JA12482-JA12491; JA12477-JA12478. Licensed embodiments of the claimed invention using indirect relationships would later revolutionize the search industry.

In 1998, Google, a licensee of the '571 Patent, was the first commercial search system to analyze citation relationships in computer databases. JA12452-JA12457. Google and third-party writers considered the unexpected results obtained from analyzing link relationships in a computer database to be a "major innovation," revolutionary, and breakthrough technology that changed the way

searches were conducted. JA12472-JA12473; JA12482-JA12491; JA12477-JA12478. Within a few years, Google's use of indirect relationships for search enabled the company to dominate the search industry. JA12469-JA12473. In contrast to the 35 and 15 year absence of any actual use after the Salton and Fox papers, within just two years after Google actually demonstrated the utility of analyzing indirect relationships, every major search engine began analyzing citations in computer databases. JA12480-JA12481. Today, 99% of the search industry has licensed the '571 Patent for substantial royalties in the multiple tens of millions of dollars. JA12492.

Critically, the evidence concerning the unexpected results does not center on the particulars of the PageRank algorithm or even whether PageRank infringes the claims. Rather, third-party commentators and Google itself note that just the very idea of using non-semantic "links" of computer databases **at all** to improve search results was novel:

> Stephen Levy [longtime technology writer and critic] describes how at the time *"**no one at the web search companies mentioned using links**. The links were the reason that a research project running on a computer in a Stanford dorm room had become the top performer."*
>
> Stanford computer science professor Rajeev Motwani describes the shift from content or text-only analysis to link analysis: "***Before this, people were only looking at the content. They were completely ignoring the fact that people were going to the effort of putting a link from one page to another and that there must be a meaning to that***."

JA12492-JA12493. In other words, the leaders of the field did not appreciate

that the non-semantic relationships of a computer database had "meaning" that could be used to enhance search. JA12467-JA12468; JA12482; JA12458-JA12468. Google's founder elaborates on the fact that the discovery that the links were "meaningful" to search was by "accident" rather than a predictable outcome of known research:

> We originally developed page rank[] kind of playing around with studying all the links on the web. And that too was a pretty ***revolutionary idea*** though it seems very simple that you could even just collect [the links of the web] and ***do anything meaningful*** …. I really credit Larry pursuing ***that idea that it is even worth collecting the graph.*** And then that you could ***run any kind of processing*** on it.

JA12377-JA12378.

> And we decided that for queries that really return a lot of results that we could do something more reasonable. ***And we sort of stumbled upon a way to do that by studying links***. . . But what we found was we—kind of ***by accident*** almost-- we found that this processing of the link structure of the web, we could create a search that was better in important ways. ***In ways that these search engines had ignored.***

JA123405. *See* JA12458-JA12468 for an extensive discussion of the unexpected results achieved by web based citation analysis.

This evidence is very important to the obviousness query regardless of whether PageRank bears a nexus with the claims or whether its success could be attributed to the patented invention. The lack of appreciation in 1998 that links in computer databases could be used under "any" algorithm to "meaningfully" improve search is independently relevant to the obviousness query. JA12467-

JA12468; JA12464; JA12458-JA12468. This lack of appreciation is fundamentally inconsistent with the idea that the widely published Fox's papers demonstrated that the claimed analysis of citations in a database was obvious. If it was a "revolutionary" idea to run "**any** kind of processing" on the non-semantic citations of a computer database to enhance search in a "meaningful" way in 1998, it certainly was not an obvious idea in 1996. JA01275-JA01277. It is clear that in 1998 the leaders of the field did not appreciate the value of analyzing hyperlink relationships of computer databases under any algorithm, despite the work of Fox. Consequently, the claimed method directed at the relationships of links on the Web is not obvious to those of ordinary skill. The comments cited above come from third party industry leaders speaking outside the context of litigation and, as such, constitute compelling evidence of nonobviousness.

When the prior art as a whole is considered for what it fairly suggests, it demonstrates that the portions of Fox's method that used *bc* and *cc* citation relationships was not useful for search. Fox simply used the wrong kind of indirect relationships (*bc* and *cc,* without other higher order relationships) in a manner that was ineffective at improving search results. JA12461-JA12462. Since his method with respect to using *bc* and *cc* was ineffective for search, a skilled artisan would not be motivated to use it to search any type of collection, much less the claimed computer database. *Id*.

6. BC and CC Alone Without Higher Order Relationships Have Never Been Proven Effective for Improving Search

Dr. Langville, award winning author on PageRank, testifies that the use of *bc* and *cc* alone without other higher order relationships would not ever have led to the discovery of the benefits of the claimed invention. JA12461-JA12462. *Bc* and *cc* are the shortest indirect relationships consisting of only two link lengths, and therefore, only capture a small amount of the link structure in a database. Daniel Egger's second numerical representation embodies an analysis of up to 18 relationships simultaneously, and its analysis includes link relationships up to four link lengths long. *Id*. Egger's "cluster link" generator and the licensed "PageRank" algorithm of Google analyzed multiple indirect relationships of any length "N." Thus, unlike Fox's method, the methods disclosed in the specification capture multiple types of empirically useful indirect relationships including higher order relationships (*i.e.*, of longer link lengths) and therefore, capture far more of the link structure of a database than just *bc* and *cc*. *Id*. To this day, no study of record has ever shown that use of *bc* and *cc* alone (without other longer citation relationships) is capable of enhancing search in a reliable way.

IV. SUMMARY OF THE ARGUMENT

Claims 12 and 22 of the '571 Patent specifically claim the analysis of hyperlink relationships on the World Wide Web and identifying websites and

URLs. JA05083(52:38-56); JA05090-JA05091(2:57-3:12). As such, these claims require one to not only recognize that indirect relationships are useful for search, but also the further inventive step of applying the analysis of these relationships to hyperlink relationships on the World Wide Web. JA12324-JA12325. As SRA's evidence demonstrates, the search industry, even after the filing of the '571 Patent, considered the analysis of hyperlink relationships on the web novel and revolutionary. JA12377-JA12378; JA12482-JA12491; JA12472-JA12473; JA12477-JA12478. The Board seized on one sentence in Fox Envision, taken out of context, to come to the opposite conclusion. JA00019-JA00020. In doing so, the Board committed clear legal errors of ignoring the empirical evidence, the teachings of the art as a whole, and clear objective indicia that the analysis of hyperlinks was nonobvious after the time of invention.

Additionally, claims 12 and 22 of the '571 Patent require the use of indirect relationships to search for objects in a database. JA05083(52:38-56); JA05090-JA05091(2:57-3:12); JA01251-JA01254. SRA provided compelling evidence that indirect relationships harm search results. Seven of nine tests performed by Dr. Fox himself demonstrated that the *bc* and *cc* indirect relationships (the only ones relied upon by Petitioners) harmed search results. JA15044-JA15045; JA12386-JA12387. No test demonstrated an improvement that met Salton's 10% threshold for significant improvement. JA12386; *supra* at 14. No test showed any

improvement by *bc* and *cc* when used as arranged in the claimed method that first created a direct relationship numerical representation. *Supra* at 14-17.

The Board completely failed to address this very important evidence significant to the obviousness inquiry. The failure to provide any reasoned explanation in its opinion as to why one would be motivated to use indirect relationships as claimed in view of the negative empirical evidence and other teachings in the art as a whole represents clear legal error. *Infra* at 41-46. Finally, the Board also failed to consider multiple objective indicia of nonobviousness. These legal errors regarding the required analysis for finding obviousness mandate that the Board's decision be set aside and given no deference. Had the Board properly conducted its analysis, it would have concluded that the claims are nonobvious.

V. ARGUMENT

A. Standard of Review

The Board's determination of obviousness is reviewed *de novo* and the factual findings for substantial evidence. *In re Gartside*, 203 F.3d 1305, 1315 (Fed. Cir. 2000). Substantial evidence is less than the weight of the evidence but more than a mere scintilla of evidence. *Id.* at 1312. A review for substantial evidence "involves examination of the record as a whole, taking into account evidence that both justifies and detracts from an agency's decision." *Id.*

Substantial evidence is "such relevant evidence as a reasonable mind might accept as adequate to support a conclusion." *Consol. Edison Co. v. N.L.R.B.*, 305 U.S. 197, 229 (1938).

Here the Board's failure to provide any analysis or explanation as to why a skilled artisan would be motivated to use indirect relationships for search when the empirical evidence and the art as a whole demonstrated such use would hurt search results represents clear legal error. As such, the Board's findings of obviousness are not entitled to deference. Indeed, the Board's stated reason for dismissing this evidence in itself represents clear legal error. *Infra* at 41-46. Conclusory findings unsupported by analysis are not entitled to deference. *In re Klein*, 647 F.3d 1343, 1350-51 (Fed. Cir. 2011).

The broadest reasonable claim construction standard, normally used in IPR proceedings, does not apply here because the Patent is expired. In construing the expired patent, the Board must apply district court claim construction principles including adopting a reasonable construction supported by the record that upholds the validity of the patent if presented with one. *Ex parte Papst-Motoren*, 1 U.S.P.Q.2d 1655, 1656 (B.P.A.I. 1986); *Ex parte Katz*, 2010 WL 1003878, at *3 (P.T.A.B. 2010); *In re Rambus Inc.*, 694 F. 3d 42, 46 (Fed. Cir. 2012) (citing *Phillips v. AWH Corp.*, 415 F. 3d 1303, 1316 (Fed. Cir. 2005)). In *Papst-Motoren*, the Board held that in case of an expired patent "a policy of liberal

construction…should be applied. Such a policy *favors a construction of a patent claim that will render it valid,* i.e., a narrow construction, over a broad construction that would render it invalid." 1 U.S.P.Q.D.2d at 1656. Here, the Board repeatedly ignored this axiom when construing the Patent.

B. The Board Erred by Concluding the Prior Art Teaches Identifying and Analyzing Hyperlinks

Claims 12 and 22 embody the inventive step of applying citation analysis to direct and indirect hyperlink relationships. JA12324-JA12325. None of the Fox Papers suggest analyzing hyperlink relationships on the World Wide Web or other networks. Hyperlinks differ from bibliographic citations in many important respects. Bibliographic citations involve an attempt to use human judgment subject to scrupulous academic review to provide a source that supports a given point. Therefore, there is an expectation that the citation may refer to related subject matter. In contrast, hyperlink references are made for many different reasons which are often commercial. They are often used for navigational purposes that do not necessarily indicate similarity of content (e.g., a link to a picture). Not only do these links have less quality as a content indicator, at the time of the Patent, they are very difficult to collect given the immense size of the internet and ever changing nature of the web and its hyperlinks. Indeed, it would be impossible to apply, to the Web, the hand methods of Fox, which took several years to index a mere 6000 articles. JA01280-JA01281.

SRA presented compelling evidence that the very idea that one can "do anything meaningful" by analyzing hyperlinks on the web for search was considered a revolutionary breakthrough in 1998. *See supra* at 25-30. Prior to that, the art and indeed the field as a whole had failed to appreciate that hyperlinks contained "meaningful" information that could be used to improve search. This lack of appreciation and the breakthrough nature of this discovery is a matter of public record by third parties outside the context of litigation and thus, constitute compelling evidence of nonobviousness.

Fox Envision does not teach the idea that one should analyze hyperlinks as opposed to bibliographic citations to enhance search. In all cases, the Fox Papers at most analyzed bibliographic citations (*i.e.*, formal citations to academic papers contained in bibliographies), and even then, Fox's results suggested that the use of bibliographic citations degraded search performance. *Supra* at 11-17. Despite the lack of any explicit teaching for this important idea that was considered years later to be a major innovation, the Board imputes this teaching based upon a misunderstanding of the following paragraph:

> We are beginning to see the emergence of wide area hypertext systems (Yankelovich, 1990) like the WorldWideWeb (WWW), that carry this concept forward into a distributed environment. ***Clearly, we must coordinate hypertext and hypermedia linking with the various approaches to search and retrieval*** ***<u>(Fox et al., 1991b).</u>*** One approach is the idea of information graphs (including hypergraphs), *where objects of all types are interrelated by links or arcs that capture not only citation (reference) but also inheritance, inclusion,*

association, synchronization, sequencing, and other relationships. JA00019-JA00020; JA05448. Nowhere, however, does this passage state that one should analyze indirect relationships as expressed in hyperlink references. The Board erroneously seizes on the emphasized portion of the above quote as teaching the steps of analyzing hyperlinks. The quote about coordinating "hypertext and hypermedia linking with the various approaches to search and retrieval" is so general that it does not suggest any particular method of search and retrieval, much less the citation analysis of Fox 1983. In fact, this statement cites to another Fox paper (Fox 1991) which is not about citation analysis at all but rather using links for navigation purposes. JA12349(¶100).

In order to find obviousness, there "must be evidence that 'a skilled artisan… would select the elements from the cited prior art references for combination *in the manner claimed*.'" *Ecolochem,* F.3d at 1375. One could "coordinate" Fox 1983 with a hypertext system or even the World Wide Web without analyzing any direct and indirect hyperlink references at all "in the manner claimed." One could place the CACM and ISI articles in a hypertext system or the web and then analyze the bibliographic citations (which are not hyperlinks) of these articles using Fox's methods described in his 1983 papers. One could use these relationships to find related text and "coordinate" hypertext to provide easy navigation between related subject matter. In this example, no direct or indirect

hyperlink relationships are ever analyzed for purposes of identifying search results. JA12349(¶100). Rather, the analysis is confined to bibliographic citations as explicitly taught in the Fox 1983 papers. Thus, the above statement relied upon by the Board does not constitute a teaching regarding the inventive step of analyzing hyperlink relationships in the manner claimed.

The above scenario is not a hypothetical but what is actually disclosed in the source code of the Fox Envision hypertext system which is the subject of Fox Envision and Fox 1991. JA12349(¶99). These files analyze the CACM database loaded into the Envision hypertext system. *Id.* The source code files from 1992 produced by Dr. Fox clearly show that the "arcs" analyzed by his system referred to above included "author", "cite", "cocite", and "bibcoup" based upon bibliographic – not hyperlink – citations. JA12349(¶100). Of course, this is fact because the articles of the CACM were created in the 1970s and do not have any hyperlinks in them to analyze. Rather, Fox analyzed the same bibliographic citations—not hyperlinks—that he analyzed in his 1983 work. Accordingly, Fox Envision's statement were not directed to link analysis of hyperlink relationships of web objects and the combination as a whole is missing this inventive step.

The Board's sole response is as follows:

> Patent Owner, in reliance on Dr. Jacobs, states that the above-referenced excerpt of Envision is not sufficient because, "[n]owhere in the referenced section[] does it say that hyperlinks or hypertext would be treated as citations for purposes of analysis." PO Resp. 18 (citing

> Ex. 2113 ¶ 92). Patent Owner and Dr. Jacobs's statements are inaccurate representations of the reference. **The approach taught in Envision is interrelating "objects of All types," including objects on the World Wide Web, so as to capture citation relationships (Ex. 1006, 482).**

JA00020. The above sentence does not say hyperlinks are analyzed as citations for searching. JA12346-JA12348. This sentence specifically teaches representing data including bibliographic citations in the form of arcs of a graph. *Id*. This means that bibliographic relationships (e.g., citations, semantic relationships, authors) can be represented as a link or arc in a graph or hypergraph.[3]

It does not mean that the arcs of the graph are derived from analyzing hyperlink references or relationships. The reference to "citations" being captured in the "links" or "arcs" of the graph refer to creating "arcs" based upon bibliographic "citations," not hyperlink ones. JA12348.

No article of Fox has ever suggested analyzing hyperlink references with citation analysis, and this was an idea he simply did not appreciate. Dr. Jacob's unrebutted review of the source code confirmed that the arcs were based upon

[3]It is should be noted that the references in the quote of a "hypergraph" does not suggest a graph of hyperlinks as the mathematical term means: "In mathematics, a **hypergraph** is a generalization of a graph in which an edge can connect any number of vertices." https://en.wikipedia.org/wiki/Hypergraph.

bibliographic citations of the CACM collection. It was impossible to be based upon anything else as the CACM collection does not have hyperlink references to other objects in it. Given the evidence that the IR field including Fox himself did not appreciate the significance of hyperlink references, this Court should not impute this missing inventive step. *See supra*. Accordingly, all claims are nonobvious.

C. The Fox Papers Do Not Teach the Identifying and Analyzing Hyperlinks as Arranged in Claim 12

Even if one assumed the Fox Papers suggested analyzing indirect relationships expressed by hyperlink citations using the Fox 1983 methods, the combination still would not teach the steps of identifying URLs that are directly related and then cluster analyzing for indirect relationships using the identified URLs to locate and identify URLs that are indirectly related. JA12328-JA12330. The 1983 Fox method does not use URLS much less identify URLs of documents (or even serving search results). Rather, the Fox 1983 method uses assigned document identifiers or "dids" to identify objects. JA05722; JA05738. These "dids" are not based upon a URL, nor do they contain any information that would otherwise identify a URL as required by the claim. Rather, the "dids" are based upon consecutive numbering that does not otherwise indicate any particular URL:

Did	Yr	Vo	No	Title (first part)	Author (first)
3184	63	01	17	Revised Report on the Algorithmic Langua	Naur, P.
3185	72	10	10	The Humble Programmer	Dijkstra, E. W.
3186	68	03	03	GO TO Statement Considered Harmful	Dijkstra, E. W.
3187	66	05	16	Certification of Algorithm 271 (QUICKERS	Blair, C. R.
3188	66	03	19	Semiotics and Programming Languages	Zemanek, H.
3189	62	11	24	An Algebraic Compiler for the FORTRAN As	Stiegler, A. D.
3190	67	02	14	Correction to Economies of Scale and the	Solomon, M.B.
3191	68	06	17	Generating Permutations by Nested Cyclin	Langdon, Glen G.
3192	58	07	02	The Lincoln Keyboard - a Typewriter Keyb	Vanderburgh, A.
3193	58	07	03	MANIAC II	
3194	59	01	02	A Non-heuristic Program for Proving Elem	Dunham, B.
3195	62	11	23	Reiteration of ACM Policy Toward Standar	Gorn, S.
3196	63	01	18	The Reactive Typewriter Program	Mooers, C. N.
3197	63	06	26	Structures of Standards-Processing Organ	Gorn, S.
3198	66	03	18	Microprogramming, Emulators and Programm	Greem, J.
3199	66	08	13	ALGEM - An Algebraic Manipulator	Gotlieb, C. C.
3200	66	08	14	A FORMAC Program for the Solution of Lin	Cuthill, E.
3201	66	08	15	Symbolic Manipulation of Poisson Series	Danby, J.
3202	66	08	16	MANIP- A Computer System for Algebra and	Bender, B.
3203	66	08	17	GRAD Assistant - A Program for Symbolic	Fletcher, J. G.
3204	66	08	18	An On-Line Program for Non-Numerical Alg	Korsvold, K.

Id. All of the steps that Petitioners rely upon for its link analysis uses these ids (raw data relations, cc and bc subvectors, and even the search results). JA12330-JA12340. Consequently, no URLs are identified when determining those objects that directly cite a selected object when creating the raw data relations or the other subvectors. After one performs Fox's method, one has no idea what the directly related URLs are of the objects that were analyzed for any specific "chosen webpage." Even when the search results are presented, the "dids" are used. Given that the Fox method can analyze citations and generate search results without any use of URLs, the steps of identifying directly related URLs and then locating URLs of indirectly related web objects are not disclosed and are completely superfluous to the disclosed method. The Board provides no rationale

or motivation that would support finding these steps obvious based upon Fox's methods disclosed in his papers.

D. The Board Further Erred by Failing to Provide an "Explanation" as to Why One Skilled in the Art Would Use Indirect Relationships to Search in View of the Empirical Evidence and Explicit Teachings that Demonstrates Indirect Relationships Harm Search Results

"[T]he question [for obviousness] is not simply whether the prior art "teaches" the particular element of the invention, but whether it would "suggest the *desirability*, and thus the obviousness, of making the combination." *Alco Standard Corp. v. Tenn. Valley Auth.*, 808 F.2d 1490, 1500 (Fed.Cir.1986). Claims 12 and 22 require using indirect relationships for search. *See supra* at 6-9. There is no more important evidence to the question of the "desirability" of using indirect relationships for search than the actual empirical results of Fox's (and other's) experiments using these relationships—those results overwhelmingly demonstrate that the use of the *cc* and *bc* indirect relationships harms search results in the claimed arrangement. *See supra* at 14-17.

SRA presented compelling evidence that demonstrated that the use of *bc* and *cc* indirect relationships in Fox's method (1) overwhelmingly harmed search results and (2) was so unpredictable that it was not fit for its intended purpose. *Id*. In rendering its decision, the Board did not address, provide weighing analysis, or otherwise make findings on the merits of these two arguments. For example, it did

not identify any flaws in SRA's interpretation of the data, nor did it identify any other evidence that contradicts SRA's claims. There is simply no analysis or other attempt to provide a reasoned explanation with respect to the merits of this very important evidence. Indeed, the opinion does not even mention SRA's evidence concerning the use of indirect relationships under Fox's method is harmful to search.

The Board's failure to provide any reasoned explanation as to why one would be motivated to use Fox's methods concerning indirect relationships given that his experimental results indicate that such use harms search results is reversible error. It is well settled that the Board *must* provide a reasoned explanation why one of ordinary skill in the art would have been motivated to select the references and to combine them in the manner claimed. *In re Kahn*, 441 F.3d 977, 986 (Fed. Cir. 2006) ("When the Board does not explain the motivation, or the suggestion or teaching, that would have led the skilled artisan at the time of the invention to the claimed combination as a whole, we infer that the Board used hindsight to conclude that the invention was obvious. … By requiring the Board to explain the motivation, suggestion, or teaching as part of its *prima facie* case, the law guards against hindsight"); *In re Rouffet*, 149 F.3d 1350,1357-59 (Fed. 1998) ("However, the Board reversibly erred…The Board provides no reasons that one of ordinary skill in this art, seeking to minimize handovers due to satellite

motion, would combine Ruddy with Rosen and King in a manner that would render the claimed invention obvious… In other words, the Board *must explain* the reasons one of ordinary skill in the art would have been motivated to select the references and to combine them to render the claimed invention obvious.”); *Apple Inc. v. Int'l Trade Com'n,* 725 F. 3d 1356 (Fed. Cir. 2013) (finding clear error when opinion fails to contain weighing analysis of important evidence, stating: “The ITC, however, never even mentioned, much less weighed as part of the obviousness analysis, the secondary consideration evidence Apple presented…This is not adequate under our law”); *Broadcom Corp. v. Emulex Corp.,* 732 F.3d 1325, 1335 (Fed. Cir. 2013) (An invention is not obvious just “because all of the elements that comprise the invention were known in the prior art;” rather a finding of obviousness at the time of invention requires a “plausible rational [sic] as to why the prior art references would have worked together”). Absent such an explanation, the Boards decision is presumed to be a product of hindsight bias and must be set aside with no deference. *Kahn*, 441 F.3d at 986.

E. The Board also Erred by Failing to Consider the Prior Art as a Whole and by Failing to Conduct a Weighing Analysis as to the Degree One Teaching May Discredit Another

It is also well established that one must consider what the prior art as a whole fairly suggests:

> ***[T]he prior art as a whole must be considered***. The teachings are to be viewed as they would have been viewed by one of ordinary skill…

> '***It is impermissible*** within the framework of section 103 ***to pick and choose from any one reference*** only so much of it as will support a given position, to the exclusion of other parts necessary to the full appreciation of what such reference fairly suggests to one of ordinary skill in the art.'

In re Hedges, 783 F.2d 1038, 1041 (Fed. Cir. 1986). Here, the Board improperly picks and chooses to rely upon Fox's teachings that some search systems support full text searching to conclude that it was obvious to apply Fox's methods to a computer database, but ignores the negative results of Fox experiments and others that demonstrate that Fox's methods concerning indirect relationships should not be used to search any collection at all, much less the claimed computer database. *Supra* at 14-17. Critically relevant to the proper inquiry is whether either the (1) Fox experiments or (2) the prior art as a whole including subsequent experimentation demonstrate that the claimed use of indirect relationships harms search results or is so unreliable that it is not useful for its intended purpose. If either of these propositions are true, then one skilled in the art would not be motivated to use the parts of the Fox combination pertaining to tests using indirect relationships, regardless if one needs to modify Fox's method or not.

Furthermore, the duty to consider the art as a whole is not confined to the teachings within one reference but rather applies across all the relevant references in the art. Indeed, the Board has a duty to weigh each reference for its suggestive power to resolve conflicts in the teaching of different art:

> When prior art contains apparently conflicting references, the Board ***<u>must</u>* weigh each reference for its power to suggest solutions to an artisan of ordinary skill**. The Board must consider all disclosures of the prior art to the extent that the references are, as here, in analogous fields of endeavor and thus would have been considered by a person of ordinary skill in the field of the invention. The Board, in weighing the suggestive power of each reference, ***must consider the degree to which one reference might accurately discredit another***.

In re Young, 927 F.2d 588, 591 (Fed. Cir. 1991). Despite the fact that the Jacobs timeline (JA12362-JA12379) puts forth substantial evidence that Fox's experimentation (as well as *experimentation by other leaders in the field)* ***demonstrated*** that indirect relationships harm search results, the Board's decision is completely devoid of the required analysis weighing the references for their suggestive power or otherwise determining the degree that one reference may discredit another, as required by the Federal Circuit. This error is particularly egregious considering the fact that among the evidence presented was that Salton, who was the thesis advisor to the Fox Papers, concluded: "the citation methodology <u>cannot be recommended</u> for inclusion in practical retrieval environments." JA12398. The Board provides no weighing analysis as to why a skilled artisan would choose to follow the alleged teaching of a graduate thesis with negative experimental results rather than the academic papers of the leaders of the IR field that explicitly and empirically determined that the use of citation relationships harm search results. JA00022-JA00023. Accordingly, the Board

clearly erred by selectively relying upon one teaching of the art without properly analyzing what the art as a whole fairly suggests.

F. Had the Board Complied with its Duty to Weigh Each Reference for its Suggestive Power to Resolve Conflicts in the Teaching in the Art, It Would Have Found All Claims Nonobvious

Had the Board properly considered the prior art as a whole, it would have been compelled to find nonobviousness. The mere fact that elements are capable of being combined is not enough to establish obviousness:

> Although predictability is a touchstone of obviousness, the "predictable result" discussed in KSR refers not only to the expectation that prior art *elements are capable of being physically combined*, but also that the ***combination would have worked for its intended purpose***…. ***The opposite conclusion would follow, however, if the prior art indicated that the invention would not have worked for its intended purpose or otherwise taught away from the invention.***

DePuy Spine, Inc. v. Medtronic SofamorDanek, Inc., 567 F.3d 1314, 1326 (Fed. Cir. 2009); *see also Eurand, Inc. v. Mylan Pharms., Inc.,* 676 F.3d 1063, 1068-69 (Fed. Cir. 2012). Thus, Petitioners had the burden to prove that searching using the indirect relationships of *bc* and *cc* is fit for its intended purpose of computerized search. The "opposite conclusion" – nonobviousness – is *compelled* if the art indicates that the combination "would not have worked for its intended purpose." *DePuy*, 567 F.3d at 1326.

Here, the Board relies upon certain statements that discuss the general capability of Fox SMART to simulate searches using *bc* and *cc*. JA01015-

JA01030. These teachings merely represent the capability to physically combine certain elements of the invention by the SMART research system—which under *DePuy* is not enough to establish obviousness. Rather, one skilled in the art would ultimately be informed on the fitness of using the *bc* and *cc* subvectors for the intended purpose of computerized search by the actual experimental results of the *bc* and *cc* subvectors reported in the performance tests of Fox Thesis. *Supra* at 11-14. Dr. Jacobs' analysis establishes that those tests indicated that the use of *bc* and *cc* would overwhelmingly result in harm to search results, and, therefore, Fox's method as to the specific data types were not fit for the intended purpose of computerized search. *Id.* Neither Petitioners nor the Board challenged the accuracy of this testimony. Jacobs further showed that no test in Fox supported the use of indirect relationships in the arrangement claimed (*i.e.*, when direct relationships representations available for use in searching). *Supra* at 14-17. In all cases, the use of *bc* and *cc* under this claimed arrangement harmed search performance.[4] **Not one test result showed an improvement over using direct links alone**. *Id.*

SRA put on direct evidence of how one skilled in the art would react to the potential deterioration of search results present in the Fox experimental results. *Supra* at 17-21. It demonstrated that when Dr. Salton was confronted with results

[4] The ISI experiments did not test direct links. JA12391(¶158).

in his experiments that contained much less degradation and far greater improvement, he still concluded that the methodology was unfit for its intended purpose: "*Overall, the procedure is not sufficiently reliable to warrant incorporation into operational automatic retrieval systems*." JA12398. SRA further showed that no person ever used indirect relationships to analyze the claimed computer databases in a non-experimental system until nearly 15 years after the Fox papers—thereby further corroborating Salton's interpretation of the evidence. *Supra* at 25-29.

Petitioners have put forward no plausible explanation that demonstrates that the use of indirect relationships would be "desirable" in view of Fox's actual test results. *Alco*, 808 F.2d at 1498 ("the question[for obviousness] is not simply whether the prior art "teaches" the particular element of the invention, but whether it would "suggest the *desirability…"*). Petitioners have failed to identify any test results that support the use of indirect relationships in a reliable fashion in the manner claimed, nor have they identified any errors in Dr. Jacobs' analysis. They have made no attempt to explain why there is a multi-decade absence of any use of *bc* and *cc* if it was obvious to use it, nor do they attempt to explain why it was considered a major discovery to analyze links in computer databases to improve search more than 15 years after the Fox papers allegedly rendered this obvious to the skilled artisan. JA12373; JA12461-JA12462.

In *Eli Lilly & Co. v. Teva Pharms. USA, Inc.*, 619 F.3d 1329, 1332 (Fed. Cir. 2010), the patent claimed the use of "raloxifene" to treat "postmenopausal osteoporosis." The defendant relied upon a reference that noted that its use suffered from "bioavailability concerns" that suggested that it would not ultimately be efficacious in humans. *Id*. at 1338-39. Despite the fact that the references explicitly tested raloxifene for treatment of bone loss (*i.e.*, the ***claimed feature)***, the Federal Circuit found nonobviousness, stating: "*[the Defendant] points to no evidence from before the time of invention that would teach, suggest, or motivate or* **supply any common sense reason** *for a person of ordinary skill in the art to* ***reject the bioavailability concerns*** *and routinely, simply, or easily arrive at the inventive result*." *Id.* at 1336-37. Just like in *Teva*, Petitioners here failed to identify any plausible explanation that would cause a person to reject the reliability concerns identified by Fox and Salton's experimental results and motivate to one routinely, simply, or easily arrive at the inventive result.

Similarly, in *Eli Lilly & Co. v. Actavis Elizabeth LLC*, the patent claimed the use of "atomoxetine" to treat ADHD. 435 F. App'x 917, 919 (Fed. Cir. 2011). The patent owner put forward evidence of "negative reports" of potential death from the use of atomoxetine; this Court held that because there was "no evidence that the advantageous and effective properties of atomoxetine to treat ADHD, *devoid of the negative effects* of known and similar products," it was impermissible to "pick

and choose from any one reference" so the claims were nonobvious. *Id.* at 921. Here, like in *Actavis*, there is no evidence or explanation of a way to take advantage of the benefits of the claimed invention without the "negative effects" of potential search degradation. Indeed, Salton explicitly states:

> Since ***no obvious way exists*** for distinguishing the positive from the ***negative effects***, the citation methodology cannot be recommended for inclusion in practical retrieval environments.

JA12398. Consequently, a finding of nonobviousness is similarly compelled.

Accordingly, when all of the teachings of the prior art are considered, Petitioners failed to meet their burden to demonstrate that a skilled artisan would be motivated to use Fox's methods as they may pertain to indirect relationships. Nor have they demonstrated that the use of *bc* and *cc* is fit for the intended purpose of computerized search. As such, the "opposite conclusion" of nonobviousness is "compelled" under *DePuy*, 567 F.3d at 1326.

G. The Board Erred by Failing to Analyze Important Objective Evidence of Nonobviousness

The Federal Circuit "has consistently pronounced that all evidence pertaining to the objective indicia of nonobviousness must be considered before reaching an obviousness conclusion," and that "[t]he significance of this fourth *Graham* factor cannot be overlooked or be relegated to 'secondary status.'" *Plantronics, Inc. v. Aliph, Inc.*, 724 F.3d 1343, 1355 (Fed. Cir. 2013). Indeed, "[o]bjective indicia 'can be the most probative evidence of nonobviousness in

the record, and enables the court to avert the trap of hindsight.'" *Leo Pharm. Prodr., Ltd. v. Rea*, 726 F.3d 1346, 1358 (Fed. Cir. 2013).

With the exception of commercial success and licensing, the Board's decision and the Petitioners' papers are devoid of any analysis of SRA's objective evidence of nonobviousness. The Board failed to address in any way the following indicators presented by SRA: lack of actual use, unexpected results (of link analysis), skepticism, failure of others, long-felt need, and praise of the industry. Under this Court's jurisprudence, all objective evidence of nonobviousness must be considered before it is legally permissible to conclude that a claim is obvious. *Ashland Oil, Inc. v. Delta Resins & Refractories*, 776 F.2d 281, 307 (Fed. Cir. 1985) (failing to consider nonobviousness evidence constitutes an "error as a matter of law."). The failure to address this evidence alone constitutes reversible error. *Transocean Offshore Deepwater Drilling, Inc. v. Maersk Drilling USA, Inc.,* 617 F.3d 1296, 1305 (Fed. Cir. 2010) (reversing the district court stating "[t]o be clear, a district court must always consider any objective evidence of nonobviousness presented in a case"). The Board's conclusion of obviousness is strongly contradicted by nearly every category of objective evidence:

Lack of Actual Use: SRA showed a complete absence of any use of any indirect relationship in a non-experimental system for 15 years after the Fox Papers and 35 years after testing on citation relationships began. *Supra* at 25-29. This

15/35 year long absence of use is strong evidence of nonobviousness. *Leo Pharm.*, 726 F.3d at 1356-59 (reversing where there was a delay of 14 years between the prior art's teachings and actual use of the claimed method—stating that the intervening time "speaks volumes to the nonobviousness of the patent.").

Unexpected Results (of link analysis): SRA provided evidence showing that the very idea of analyzing citation relationships in a computer database to improve search was not appreciated until nearly 15 years after the Fox Papers. Leaders of the field expressed surprise in 1998 that the hyperlink citation links in web documents could be used in "any meaningful" way. JA12467-JA12468; JA12464; JA12458-JA12468. If the appreciation of analyzing computer database relationships, such as hyperlink citations, to improve search was "revolutionary" in 1998 to the leaders of the industry, it could not have been rendered obvious to those of ordinary skill by Fox in 1983.

Skepticism by Experts: Jacobs's testimony showed that the prior art as a whole overwhelmingly discouraged the use of indirect relationships. *Supra* at 21-25. Dr. Salton explicitly noted that the "***initial reaction must clearly be one of <u>skepticism</u>***" toward the use of bibliographic citations. JA12363. Experts as late as 1992 (the year before the filing of patents) were criticizing the Fox methodology as being unreliable as well as noting that analysis of citations produces "ambiguous results." JA12375-JA12376. No less than four empirical studies (Salton '63, Fox

’83, Salton ’86 and Nunn ’87) largely demonstrated that indirect relationships harm search results and failed to prove that citations can be used in a reliable way. JA15074. No study of record empirically demonstrates that *bc* and *cc* alone can be used to improve search in a reliable way. *Kinetic Concepts, Inc. v. Smith & Nephew, Inc.,* 688 F.3d 1342, 1367 (Fed. Cir. 2012) (invention properly found nonobvious where "leading experts in the field were skeptical that the [it] could work"); *In re Dow Chem. Co*., 837 F.2d 469, 473 (Fed. Cir. 1988) (reversing PTO finding of obviousness where experts were skeptical despite five to six years of research).

Failure of Others/Long-Felt Need: Jacobs’s timeline establishes a 40-year period beginning with Salton ’63 where the art of record discusses the need for improvement of search results began testing potential citation solutions to the problem. JA12362-JA12379. Despite nearly 40 years since testing began, the prior art as a whole indicates a field that struggled with finding any empirically supported use of indirect relationships that could be fashioned into a successful search method until well after the patents-in-suit. JA12362-JA12379; JA12493-JA12495. SRA provided a solution to this long-felt problem by analyzing multiple types of empirically useful indirect relationships (18 different patterns and higher order relationships of length “N”) that capture more of the link structure than the tested patterns of *bc* and *cc* (focused on by the prior art) and therefore, reliably

improved search. JA12461-JA12462. This testimony was unrebutted. Such unrebutted testimony shows that the invention was a nonobvious solution. *Alco*, 808 F.2d at 1500-01 (finding "strong secondary considerations indicating nonobviousness" where the evidence showed that "'for well over a decade the industry had searched for a reliable method of detecting discontinuities in rotor forgings,' [and] (b) that 'major turbine manufacturers had tried and failed to develop a reliable method'"); *Eurand*, 676 F.3d at 1081 ("'[T]here can be little better evidence negating an expectation of success than actual reports of failure….' In such circumstances, 'evidence of failed attempts by others could be *determinative* on the issue of obviousness.'"). The repeated failures of Fox, Salton, and others in the art to develop an empirically effective search method using indirect relationships is compelling evidence of nonobviousness.

Similarly, the analysis of link relationships of a computer database to enhance search was another solution to the long felt need to improve search ranking. JA12493-JA12495. This solution was ignored until SRA's licensees demonstrated its utility in 1998 and revolutionized the search industry. All major search engines now license and use Egger's solution. *Supra* at 27.

Industry Acquiescence: SRA has showed industry acquiesce to the validity of the patents based upon pervasive licensing of the patents. 99% of the search industry have taken licenses under the Patent for over $30 million, with at least

half of that amount being paid by Google. JA12492. This widespread licensing for millions of dollars supports the nonobviousness of the claimed invention. *See, e.g., Institut Pasteur & Universite Pierre et Marie Curie v. Focarino,* 738 F.3d 1337, 1347 (Fed. Cir. 2013) (licensing of the patent is "probative and cogent evidence" of nonobviousness). Despite the large sums of money being paid, which indicate merit of the claims, the Board erroneously dismissed this evidence because the licenses were done in the context of litigation. *Rambus Inc. v. Rea*, 731 F.3d 1248, 1254-55, 1257 (Fed. Cir. 2013) (reversing the Board's finding that Rambus's licensing evidence lacked a nexus because competitors may take licenses for reasons including litigation).

Unexpected Results, Praise and Commercial Success of PageRank: SRA put on substantial evidence that PageRank, a licensed embodiment, produced unexpected results that "revolutionized" the search industry. JA12851-JA12861. Further, Google's dramatic takeover of the search industry using the licensed PageRank points to the nonobviousness of the claimed invention. JA12458-JA12468. Further, Google's dramatic takeover of the search industry using the licensed PageRank points to the nonobviousness of the claimed invention. JA12469-JA12472. Although the Board did not find commercial success, it made a number of errors that caused to reach the wrong conclusion. *Infra* at 56-63.

Each of the above objective indicators is irreconcilable with the Petitioner's position that the Fox experiments rendered the use of indirect relationships for search obvious. Each of these indicators, however, confirm and corroborate Dr. Jacobs' testimony that the Fox methods regarding *bc* and *cc* harmed search results and were considered too unreliable for use in computerized search. *Supra* at 11-21. When the experimental results of Fox are considered in the context of the prior art as a whole and the objective evidence of nonobviousness, it is clear that the Board erred in its ultimate legal conclusion of obviousness, and this Court should find the claims nonobvious.

H. The Board Did Not Properly Consider SRA's Prima Facie Case of Nexus

The Board contends that Appellant did not show commercial success because:

> As explained in the Declaration of Dr. Jacobs, Google's "legendary insight was to rate pages based on the number and importance of links that pointed to them." … Nonetheless, Dr. Jacobs statement regarding Google's technology applies to direct citations. Also, Patent Owner's characterization of Google's technology indicates that "importance" of links is used to rate pages, which is too vague to be considered as persuasive evidence of a nexus between Google's technology and claim 12.

JA00024. The Board dismisses Jacob's testimony as to the "number and importance of links" as being too vague to establish a nexus to the claims that require an analysis of indirect relationships. *Id.* The Board apparently missed the

declaration of Dr. Langville, award winning author on PageRank, which sets forth (1) that Google and third parties identified PageRank as <u>the</u> reason for its success (JA12482-JA12489; JA12463-JA12462); (2) highly detailed claim charts based on Google's publicly disclosed algorithms showing element-by-element how the use of PageRank practices the claimed invention (JA11111-JA11285), (3) testimony that she reviewed the PageRank code and it conformed with Google's disclosed algorithm (JA12457(¶39)), (4) Google paid an eight digit license fee to SRA (JA12492), and (5) 99% of the search industry paid fees to SRA for the use of "PageRank." (JA12492). This evidence is not hearsay as the Board contends because under 37 C.F.R. §42.53(a) the Langville Declaration constitutes direct testimony for which Petitioners had an opportunity to cross examine. Experts may also rely upon hearsay testimony. FRE 703. Indeed, the Board decision does not even mention this testimony concerning commercial success in its decision.

As shown by its disclosed algorithm, PageRank analyzes hyperlink citation relationships, rather than words, and, consequently, is a purely non-semantical analysis. JA11289; JA11950; JA12454-JA12455. Although PageRank itself does not contain any semantical analysis, it is used in conjunction with other semantical factors of the Google search engine. JA12229 ("each page[] has a popularity score [PageRank], which is computed *solely* from the structure index…. This popularity

score is then combined with the content score [semantic factors]… to create an overall score for each relevant page.”).

Langville’s 75 page declaration also clearly establishes that PageRank was the reason for Google’s success. JA12482-JA12489; JA12463-JA12462. Both Google and third party observers have *repeatedly* and publically attributed *the reason* for their success to PageRank:

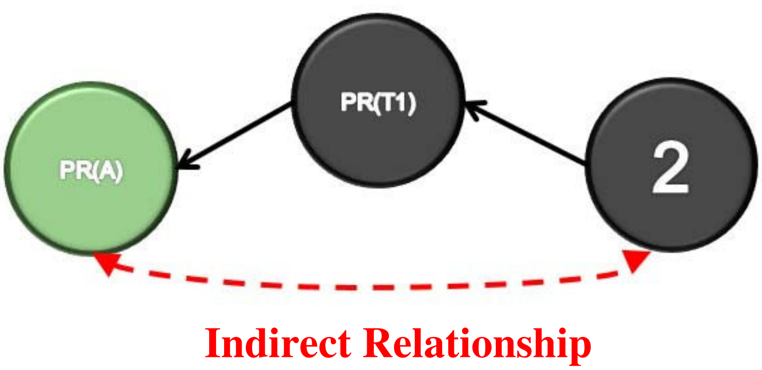
> ***“The reason why my system works so well*** is that it decides which documents to return, and in what order, by using an approximation to how well cited or ‘important’ the matching documents are. I will call this approximation to importance ***PageRank*** from now on.” (Lawrence Page, CEO Google).
>
> “The approach used by the search engines in 1996 [i.e., semantical search] would not have gotten us to today. They would not have gotten us here. ***What got us here was an insight that Larry, mostly Larry, but Larry and Sergey had together called PageRank.”*** (Douglas Merril, Google VP of Engineering).
>
> “PageRank was the backbone of the Google success.”
>
> “Google's breakthrough in search, which quickly made it the undisputed search market leader, was PageRank.”
>
> “Google [is] built around Page and Brin’s breakthrough PageRank algorithm. Even more telling, an estimated 99% of its profit [is] too.”

Id. Google clearly is the best judge of the cause of their own success. This was unrebutted by Petitioners.

Second, the Board erroneously concluded that the PageRank algorithm only “applies to direct citations.” JA00024. Had the Board evaluated Langville’s

testimony, it would have seen detailed claim charts showing, on a limitation-by-limitation basis, infringement by Google's use of PageRank. JA11111-JA11285. Dr. Langville testifies that the algorithm disclosed in their publications, patents and source code set forth in the claim charts is a recursive analysis of indirect relationships on the World Wide Web and not merely an analysis of direct citations as suggested by the Board:

$$PR(A) = (1\text{-}d) + d\ (PR(T1)/C(T1) + ... + PR(Tn)/C(Tn))$$



$$r(A) = \frac{\alpha}{N} + (1-\alpha)\left(\frac{r(B_1)}{|B_1|} + \cdots + \frac{r(B_n)}{|B_n|}\right),$$

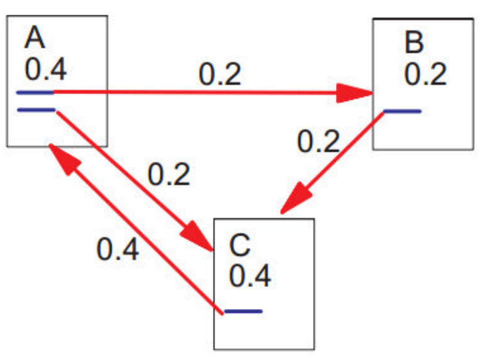


Figure 3: Simplified PageRank Calculation

JA11228-JA11234. Contrary to the Board's opinion, these charts are not based on vague evidence, but, instead, rely on publications by Google's founders disclosing Google's PageRank algorithm. This evidence was not addressed by the Board. Nor was the fact that it was directed to indirect relationships even challenged by the Petitioners.

Furthermore, Langville testified that Google and every other search engine using PageRank obtained a license under the '571 Patent for its use. JA12492. She testified that every asserted claim of the '571 Patent required an analysis of indirect relationships. JA11219-JA11285. Google paid an eight digit settlement which is far above nuisance value. JA12492. This acquiesce to a large settlement further corroborates Langville's assertion that the use of PageRank analyzes indirect relationships and infringes the '571 Patent.

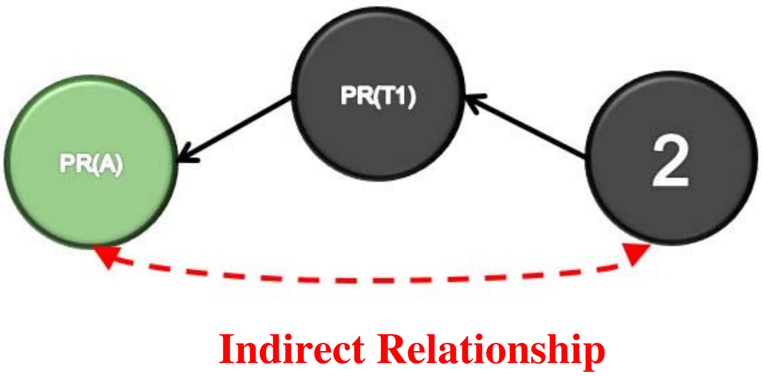
The Jacobs Declaration quote does not suggest that PageRank is only limited to an analysis of direct relationships. Google itself explains how the "importance of a link" is determined by recursively analyzing PageRank of the nodes that link to a node that is linked to the node being PageRanked (i.e., indirect relationship):

> PageRank is a query-independent technique for determining the importance of web pages by looking at the link structure of the web. PageRank treats a link from web page A to web page B as a "vote" by page A in favor of page B. **The PageRank of a page is the sum of the PageRank of the pages that link to it. The PageRank of a web page also <u>depends on the importance (or PageRank) of the other web pages casting the votes</u>.** Votes cast by important web pages with

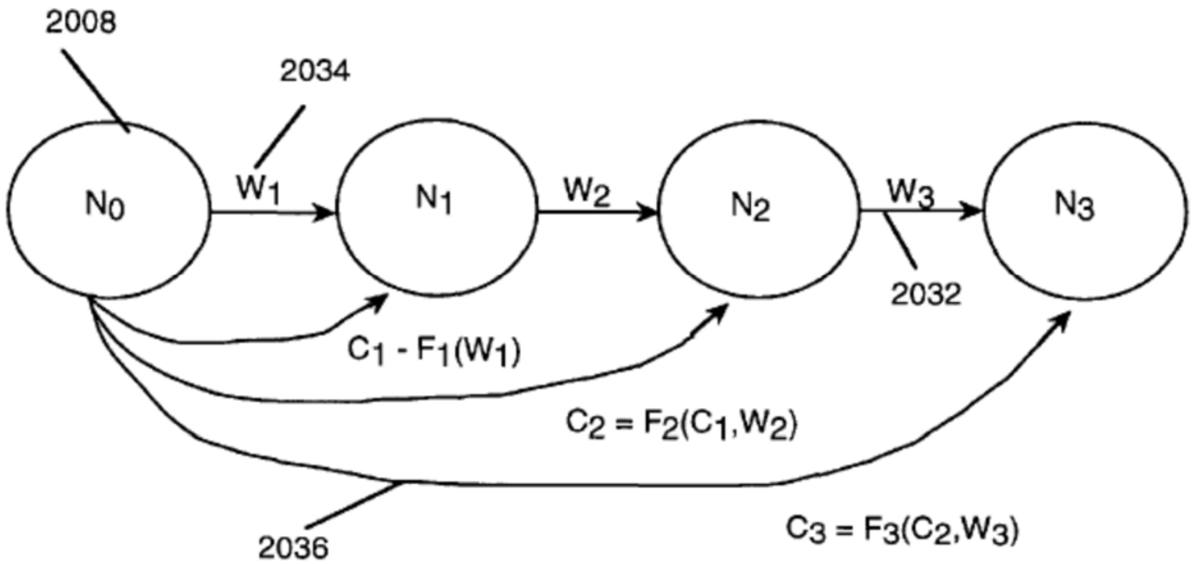
> high PageRank weigh more heavily and are more influential in deciding the PageRank of pages on the web.

JA11344. For example, assume page 2 points to page T1, which points to page A, then there is an indirect relationship between page 2 and page A:

$$PR(A) = (1\text{-}d) + d\ (PR(T1)/C(T1) + \ldots + PR(Tn)/C(Tn))$$



JA11228-JA11234. The PageRank score for page A is based on the PageRank score for page T1, and the PageRank score for link T1 is based on the PageRank score for page 2. Thus, the PageRank score for page A is defined by the claimed analysis of the indirect hyperlink relationship between page 2 and page A. *Id.* Note that this recursive algorithm is substantially similar to the recursive cluster analysis identified in Fig. 3G which defines a link's score C_3 by the score C_2 of the node linked to it:



JA05009. Accordingly, the rating of page's "importance of links that pointed to them," as described by Dr. Jacobs, requires an analysis of ***indirect relationships, a claimed feature***.

Accordingly, SRA met its burden of establishing a prima facie case by showing that PageRank was the reason for Google's success and that its use falls within the ambit of the claims. *See Crocs, Inc. v. ITC,* 598 F.3d 1294, 1311 (Fed. Cir. 2010). In contrast, Petitioners provided no rebuttal evidence. Instead, Petitioners generally assert, without any supporting evidence, that Langville failed to consider other factors that *may* have caused the success of Google. JA01364-JA01368. These unsupported arguments speculating that factors other than PageRank may have contributed Google's success are simply insufficient in view of the prima facie case of SRA. Petitioners failed to provide the requisite evidence

to establish a "convincing case" that these factors were *in fact the likely cause* of the success of Google. *See Crocs, Inc.*, 2008-1596 at 24 ("Once the patentee demonstrates a prima facie nexus," the defendants "must make a convincing case that those market forces indeed were the likely cause of the success.") *Id*. at 24. Accordingly, the Board erred in its conclusion that claims do not bear a nexus with PageRank.

CONCLUSION

For these reasons, the Court should find claims 12 and 22 nonobvious.

Dated: July 27, 2015

Respectfully submitted,
/s/Minghui Yang
Martin M. Zoltick
Rothwell, Figg, Ernst & Manbeck, P.C.
607 14th Street, N.W., Suite 800
Washington, DC 20005

Victor Hardy
Minghui Yang
DiNovo Price Ellwanger & Hardy LLP
7000 North MoPac Expy, Suite 350
Austin, TX 78731

Attorneys for Appellants

CERTIFICATE OF SERVICE

I hereby certify that on this 27th day of July, 2015, the foregoing BRIEF OF APPELLANT was filed electronically with the U.S. Court of Appeals for the Federal Circuit by means of the Court's CM/ECF system. I further certify that the foregoing was served by means of electronic mail as well as by the Court's CM/ECF system, which should have sent a Notice of Docket Activity, upon the following counsel of record for Cross-Appellants:

Heidi L. Keefe, Esq.
Mark R. Weinstein, Esq.
Cooley LLP
1299 Pennsylvania Ave., N.W., Suite 700
Washington, DC 20004
Ph: 650-843-5001
Fx: 650-849-7400
E-mail: hkeefe@cooley.com
mweinstein@cooley.com

Counsel for Appellee Facebook, Inc.

David Silbert, Esq.
Asim M. Bhansali, Esq.
Sharif E. Jacobs, Esq.
Keker & Van Nest LLP
633 Battery Street
San Francisco, CA 94111
Ph: 415-391-5400
Fx: 415-397-7188
E-mails: djs@kvn.com
abhansali@kvn.com
sjacob@kvn.com

Counsel for Appellees LinkedIn Corp. and Twitter, Inc.

/s/Minghui Yang
Minghui Yang
DiNovo Price Ellwanger & Hardy LLP
7000 North MoPac Expressway
Suite 350
Austin, Texas 78731
Telephone: 512-539-2626
Facsimile: 512-539-2627
E-mail: myang@dpelaw.com

CERTIFICATE OF COMPLIANCE

1. This brief complies with the type-volume limitation of Federal Rule of Appellate Procedure 32(a)(7)(B) because this brief contains 13,607 words, excluding the parts of the brief exempted by Federal Rule of Appellate Procedure 32(a)(7)(B)(iii).

2. This brief complies with the typeface requirements of Federal Rule of Appellate Procedure 32(a)(5) and the type-style requirements of Federal Rule of Appellate Procedure 32(a)(6) because this brief has been prepared in proportionally spaced typeface using Microsoft Word in 14-point Times New Roman font.

Date: July 27, 2015

Respectfully submitted,

/s/Minghui Yang

Minghui Yang
DiNovo Price Ellwanger & Hardy LLP
7000 North MoPac Expressway
Suite 350
Austin, Texas 78731
Telephone: 512-539-2626
Facsimile: 512-539-2627
E-mail: myang@dpelaw.com

ADDENDUM 1

Final Written Decision – 35 U.S.C. § 318(a) and 37 C.F.R. § 42.73,
Facebook, Inc. et al. v. Software Rights Archive, LLC, Case IPR2013-00481
(P.T.A.B. January 29, 2015) (Paper 54)

JA00001 – JA00037

2015-1652, 2015-1653

SOFTWARE RIGHTS ARCHIVE, LLC
v.
FACEBOOK, INC., LINKEDIN CORPORATION, TWITTER, INC.

Trials@uspto.gov
571.272.7822

Paper 54
Entered: January 29, 2015

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

FACEBOOK, INC., LINKEDIN CORP., and TWITTER, INC.,
Petitioner,

v.

SOFTWARE RIGHTS ARCHIVE, LLC,
Patent Owner.

Case IPR2013-00481
Patent 6,233,571 B1

Before SALLY C. MEDLEY, CHRISTOPHER L. CRUMBLEY, and BARBARA A. PARVIS, *Administrative Patent Judges.*

PARVIS, *Administrative Patent Judge.*

FINAL WRITTEN DECISION
35 U.S.C. § 318(a) and 37 C.F.R. § 42.73

I. BACKGROUND

A. Introduction

On July 30, 2013, Facebook, Inc., LinkedIn Corp., and Twitter, Inc. (collectively "Petitioner") filed a petition ("Pet.") requesting an *inter partes* review of claims 12, 21, 22, 26, 28, and 31 of U.S. Patent No. 6,233,571 B1 (Ex. 1001, "the '571 Patent"). Paper 1. On February 3, 2014, we instituted

trial for challenged claims 12, 21, and 22 of the '571 Patent on certain of the grounds of unpatentability alleged in the Petition. Paper 16 ("Decision to Institute" or "Inst. Dec."). We denied institution of *inter partes* review of claims 26, 28, and 31 of the '571 Patent. *Id*.

After institution of trial, Patent Owner, Software Rights Archive, LLC ("Patent Owner"), filed a Patent Owner Response ("PO Resp."). Paper 29. Petitioner also filed a Reply. Paper 38 ("Reply").

A consolidated oral hearing for IPR2013-00478, IPR2013-00479, IPR2013-00480, and IPR2013-00481, each involving the same Petitioner and the same Patent Owner, was held on October 30, 2014. The transcript of the consolidated hearing has been entered into the record. Paper 53 ("Tr.").

We have jurisdiction under 35 U.S.C. § 6(c). This final written decision is issued pursuant to 35 U.S.C. § 318(a) and 37 C.F.R. § 42.73.

Petitioner has shown by a preponderance of the evidence that claims 12 and 22 of the '571 Patent are unpatentable.

Petitioner has not shown by a preponderance of the evidence that claim 21 of the '571 Patent is unpatentable.

B. Related Proceedings

Petitioner indicates that the '571 patent is involved in the following co-pending lawsuits: *Software Rights Archive, LLC v. Facebook, Inc.*, No. 12-cv-3970 (N.D. Cal., filed July 27, 2012); *Software Rights Archive, LLC v. LinkedIn Corp.*, No. 12-cv-3971 (N.D. Cal., filed July 27, 2012); and *Software Rights Archive, LLC v. Twitter, Inc.*, No. 12-cv-3972 (N.D. Cal., filed July 27, 2012). Pet. 2. Petitioner also indicates that the '571 patent was the subject of prior lawsuit: *Software Rights Archives, Inc. v. Google*, No. 08-cv-03172 (N.D. Cal.) ("Google Lawsuit"). Pet. 7–8.

The ’571 patent was the subject of Reexamination 90/011,012. Additionally, Petitioner filed other petitions on related patents including: IPR2013-00478, which seeks *inter partes* review of U.S. Patent No. 5,544,352 (“the ’352 Patent”) and IPR2013-00479 and IPR2013-00480, each of which seeks *inter partes* review of U.S. Patent No. 5,832,494 (“the ’494 Patent”). The ’571 Patent issued from an application that was a divisional of the application that issued as the ’494 Patent. The ’352 Patent issued from the parent of the application that issued as the ’494 Patent.

C. The ’571 Patent

The ’571 Patent relates to computerized research on databases. Ex. 1001, 1:16–17. According to the ’571 Patent, it improves search methods by indexing data using proximity indexing techniques. *Id*. at 3:25–36. The ’571 Patent further states that its computerized system for researching data is effective for indexing and searching the Internet and the World Wide Web. *Id*. at 48:19–26.

D. Illustrative Claims

The independent claims are 12, 21, and 22.

Independent claims 12 and 21 illustrate the claimed subject matter and are reproduced below:

> 12. A method for visually displaying data related to a web having identifiable web pages and Universal Resource Locators with pointers, comprising:
>
> choosing an identifiable web page;
>
> identifying Universal Resource Locators for the web pages, wherein the identified Universal Resource Locators either point to or point away from the chosen web page;
>
> analyzing Universal Resource Locators, including the identified Universal Resource Locators, wherein Universal

Resource Locators which have an indirect relationship to the chosen web page are located, wherein the step of analyzing further comprises cluster analyzing the Universal Resource Locators for indirect relationships; and

displaying identities of web pages, wherein the located Universal Resource Locators are used to identify web pages.

21. A method of displaying information about a network that has hyperjump data, comprising:

choosing a node;

accessing the hyperjump data;

identifying hyperjump data from within the accessed hyperjump data that has a direct reference to the chosen node;

determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node using the identified hyperjump data, wherein the step of determining comprises [cluster analyzing the hyperjump data] *non-semantically generating a set of candidate cluster links for nodes indirectly related to the chosen node using the hyperjump data, assigning weights to the candidate cluster links and deriving actual cluster links from the set of candidate cluster links based on the assigned weights*; and

displaying one or more determined hyperjump data.[1]

E. The Prior Art References Supporting Alleged Unpatentability

Edward A. Fox, et al., *Users, User Interfaces, and Objects: Envision, a Digital Library*, 44 J. AM. SOC. INF. SCI., no. 8 at 480–91 (Sept. 1993) ("Envision") (Ex. 1006).

Edward A. Fox, *Extending the Boolean and Vector Space Models of Information Retrieval with P-Norm Queries and Multiple Concept Types*,

[1] Claim 21 is shown as it appears in the Reexamination Certificate.

(Aug. 1983) (Ph.D. dissertation, Cornell Univ. Dept. of Comp. Sci.) ("Fox Thesis") (Ex. 1012).

Edward A. Fox, *Some Considerations for Implementing the SMART Information Retrieval System under UNIX*, (Sept. 1983) (Ph.D. dissertation, Cornell Univ. Dept. of Comp. Sci.) ("Fox SMART") (Ex. 1013).

The parties do not dispute the prior art status of the references.

F. *The Pending Ground of Unpatentability*

References	Basis	Claims challenged
Fox Thesis, Fox SMART, and Envision	§ 103	12, 21, and 22

II. ANALYSIS

A. Claim Construction

1. Principles of Law

Petitioner asserts, and Patent Owner does not dispute, that the '571 Patent expired on June 14, 2013. Pet. 7. The Board's interpretation of the claims of an expired patent is similar to that of a district court's review. *See In re Rambus, Inc.*, 694 F.3d 42, 46 (Fed. Cir. 2012). We, therefore, are guided by the principle that the words of a claim "are generally given their ordinary and customary meaning," as understood by a person of ordinary skill in the art in question at the time of the invention. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–13 (Fed. Cir. 2005) (en banc) (citation omitted). "In determining the meaning of the disputed claim limitation, we look principally to the intrinsic evidence of record, examining the claim language itself, the written description, and the prosecution history, if in evidence."

DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc., 469 F.3d 1005, 1014 (Fed. Cir. 2006) (citing *Phillips*, 415 F.3d at 1312–17). There is a "heavy presumption," however, that a claim term carries its ordinary and customary meaning. *CCS Fitness, Inc. v. Brunswick Corp.*, 288 F.3d 1359, 1366 (Fed. Cir. 2002) (citation omitted).

2. Overview of the Parties' Positions

In the Decision to Institute, we construed "non-semantically," "cluster links," "candidate cluster links," "Universal Resource Locators," and "web page." Inst. Dec. 9–11. Our constructions are set forth in the table below.

Claim Term or Phrase	Construction
"non-semantically"	"[B]ased on direct relationships between textual objects and that otherwise does not account for phrases and words in a textual object." Inst. Dec. 9.
"cluster links"	"[R]elationships used for grouping interrelated nodes." Inst. Dec. 10.
"candidate cluster links"	"[A] set of possible cluster links between a search node and a target node." Inst. Dec. 11.
"Universal Resource Locators"	"[A]n address commonly used for a web page." Inst. Dec. 11.
"web page"	"[A] document on the World Wide Web." Inst. Dec. 11.

Petitioner does not disagree with any of our constructions. Reply 2. Patent Owner appears to agree with many of our constructions, and states that it uses our constructions for the purpose of evaluating patentability of the challenged claims of the '571 patent. PO Resp. 12–13. Patent Owner, however, expressly disagrees with our construction of "cluster links" and provides an alternate construction (*id*. at 13), which we evaluate below.

Patent Owner additionally contends that "[a]ll claims require the use of an analysis of indirect relationships for computerized searching," *Id*. at 2, so, in addition to "cluster links," we also evaluate our constructions of "indirect relationships" and "indirect reference" in claims 12 and 22 respectively.

In addition to these express disputes, Patent Owner also makes arguments in its response that implicitly rely on narrower constructions than we adopted in our Decision to Institute. For example, Patent Owner makes arguments based on a construction of "web page," as recited in claim 12, that is narrower than the construction that we adopted in the Decision to Institute. PO Resp. 21–22. Patent Owner further makes arguments based on a specific order of steps of claim 21. *Id.* at 48–49. We consider these disputes to be claim construction issues and consider them below.

Additionally, Patent Owner provides proposed constructions for "proximity indexing" and "cluster analyzing," which are evaluated below. PO Resp. 13–14. Petitioner does not contest Patent Owner's proposed constructions of these terms. Reply 2.

With the exception of the terms that we note as requiring additional evaluation, for each of the other claim terms above, we discern no reason, based on the complete record now before us, to change our construction thereof.

3. "cluster links"

The term "cluster links" is recited in claim 21. Patent Owner asserts that the correct interpretation of "cluster links" is narrower than the construction that we adopted. *See e.g.*, PO Resp. 13. Patent Owner's contentions are based on two alternate constructions of "cluster links." First, according to Patent Owner, "cluster link" means: "a relationship between

two nodes based [upon] statistical analysis of multiple relationships between [] nodes in a database." *Id.* (citing Ex. 2021).

Second, Patent Owner contends that "[a]ll claims require the use of an analysis of indirect relationships for computerized searching." PO Resp. 2. Patent Owner contends that the asserted art, in contrast, simply teaches "experiments to determine whether direct and indirect relationships existing <u>among paper documents</u> are useful for clustering and searching." *Id.* at 1.

Regarding Patent Owner's first contention, Patent Owner does not explain why the asserted construction is correct. *See* 37 C.F.R. §§ 42.23, 42.120. Patent Owner, instead, cites to the analysis in Exhibit 2021. Patent Owner, however, does not submit Exhibit 2021. *See* Patent Owner's Exhibit List (Paper 31), 3.

Petitioner submits a Claim Construction Order in the Google Lawsuit ("Claim Construction Order," Ex. 1011), which adopts the construction above. The construction was argued by defendants in the Google Lawsuit, which are not parties to this *inter partes* review. The Claim Construction Order includes analysis of the specification of the '494 Patent and, in particular, a document incorporated by reference in the '494 Patent specification, referred to as "V-Search Manual." Ex. 1011, 15–18. Patent Owner has not pointed us to citations in the V-Search Manual and has not identified this manual within the record of this *inter partes* review. Thus, we decline to adopt Patent Owner's first proposed construction.

Regarding Patent Owner's second contention that the challenged claims require the use of an analysis of indirect relationships for computerized searching (PO Resp. 2), we evaluate whether our prior construction of cluster links should be modified to clarify that the

relationships must be represented in data stored in a computer, not existing merely among printed documents.

Claim 21 recites a method of displaying information about a network, which includes a determination regarding hyperjump data. As explained in the ’571 Patent specification, exemplary hyperjump data includes hyperjump links, which connect web pages, web sites, and documents on the web. Ex. 1001, 48:37–39. The specification of the ’571 Patent states that a link is a “relationship between two nodes.” *Id.* at 13:5–6. The ’571 Patent specification continues, “[a] link [] can be represented by a vector or an entry on a table and contain information for example, a from-node identification [] (ID), a to-node ID [], a link type [], and a weight.” *Id*. at 13:9–12. As described in the ’571 Patent specification, a link is represented in data stored in a computer.

We, therefore, construe “cluster links” in light of the specification of the ’571 Patent to mean relationships, which are represented in data stored in a computer and are used for grouping interrelated nodes.

4. “indirect relationships” and “indirect reference”

The term “indirect relationships” is recited in claim 12 and the term “indirect reference” is recited in claim 22. As discussed above, the terms “link” and “relationship” are used interchangeably in the ’571 Patent specification. For the reasons discussed above, therefore, we construe “indirect relationships” to mean indirect relationships that are represented in data stored in a computer. For these same reasons, we construct “indirect reference” to mean an indirect reference that is represented in data stored in a computer.

5. “proximity indexing”

The term “proximity indexing” is recited, for example, in claim 22. Patent Owner argues that “proximity indexing” means “techniques [that] generate a quick-reference of the relations, patterns, and similarity found among the data found in the database.” PO Resp. 13–14 (citation omitted). Petitioner does not contest this construction. Reply 2.

As an initial matter, Patent Owner has not explained persuasively how one of ordinary skill in the art would understand what types of relations are “quick-reference” and what “similarity among data” means.

Additionally, Patent Owner’s proposed construction refers to certain types of output generated by proximity indexing techniques. As explained by the ’571 Patent specification, proximity indexing involves indexing many types of data including “very large databases” (Ex. 1001, 27:24), documents in databases of “law firms, businesses, government agencies, etc.” (*id*. at 27:43–46), and “shapes” (*id*. at 27:63), which can be used to “compare line drawings of known pottery to a newly discovered archeological find” (*id*. at 28:15–16) and “scan through and compare police composite drawings” (*id*. at 28:17–18), as well as perform other searches (*id*. at 28:12–22).

Regarding defining the term “proximity indexing,” the specification of the ’571 Patent states “Proximity Indexing is a method of preparing data in a database for subsequent searching.” Ex. 1001, 3:59–60. The ’571 Patent specification further describes proximity indexing with respect to a program that “indexes (or represents) data in a locally located database or a remotely located database.” *Id*. at 4:5–7; *see also id*. at 3:66–67 (“The Proximity Indexing Application Program indexes (or represents) the database in a more useful format . . . to efficiently search the database.”)

The specification of the ’571 Patent provides additional specificity regarding proximity indexing including that it “organize[s] and categorize[s] data stored in databases.” Ex. 1001, 13:41–43. The specification of the ’571 Patent, furthermore, describes proximity indexing as indexing data “based on their degree of relatedness . . . to one another.” *Id*. at 13:47–50.

The ’571 Patent specification also provides exemplary types of proximity indexing, including generating cluster links. In particular, the ’571 Patent specification states, “[t]he following describes a preferred cluster link generator [] which implements a specific type of patter[n]er or clustering system for use along or in conjunction with *other proximity indexing subroutines*.” *Id*. at 21:30–33 (emphasis added); *see also id*. at 4:20–22 (“The Proximity Indexing Application Program then *clusters* related contiguous paragraphs into sections.”) (emphasis added).

Upon review, we construe “proximity indexing,” in light of the specification of the ’571 Patent to mean preparing data in a database for subsequent searching by organizing and categorizing the data based on their degree of relatedness to one another. Additionally, we determine that “proximity indexing” encompasses examples set forth in the specification of the ’571 Patent including generating a set of candidate cluster links.

6. “cluster analyzing”

The term “cluster analyzing” is recited, for example, in claim 12. Patent Owner contends that “cluster analyzing” at least involves generating cluster links. PO Resp. 14. Petitioner does not contest Patent Owner’s contention. Reply 2.

Patent Owner supports its contention by relying on its Declarant, Dr. Paul S. Jacobs. PO Resp. 14 (citing Ex. 2113 ¶ 51). Dr. Jacobs states that

Figure 14B of the ’571 Patent specification illustrates a flow chart for performing “cluster analyzing,” as recited in claim 12. Ex. 2113 (citing Ex. 1001, 49:38–40). The text of the ’571 Patent on which Dr. Jacobs relies, however, does not state that Figure 14B is a flow chart for performing cluster analyzing. Instead, the ’571 Patent states, “FIG. 14B describes the embodiment of the invention which executes 3020 the cluster link generator algorithm 2044 to generate direct and indirect links 2004 to find the set of candidate cluster links.” Ex. 1001, 49:37–40; *see also id*. at 10:5–7 (“FIG. 14B is a high level diagram of a method for searching, indexing, and displaying data stored in a network using the cluster generating algorithm.”)

Figure 3A of the ’571 Patent describes an “overall procedure” (Ex. 1001, 16:37) for an algorithm that indexes and formats data (*id*. at 16:33–38). Figure 3A includes a step that simply states “CLUSTER AND SECTION.” Ex. 1001, Fig. 3A. The specification of the ’571 Patent also describes the “real power” of the indexing algorithm of the patent as “allow[ing] one to identify ‘groups’ or ‘clusters’ of interrelated cases.” *Id*. at 16:26–28. The ’571 Patent specification additionally indicates that its indexing algorithm used with the World Wide Web is not limited to the algorithm of Figure 14B, “[t]his computerized system for researching data is also effective with any type of internal or global network application (see generally FIGS. 14A and 14B).” *Id*. at 48:19–21; *see also id*. at 21:30–33 (“The following describes a preferred cluster link generator 2044 which implements a specific type of patter[n]er or clustering system for use alone or in conjunction with other proximity indexing subroutines.”)

Upon review, we construe "cluster analyzing," in light of the specification of the '571 Patent to mean identifying clusters. We do not agree that "cluster analyzing" involves at least generating cluster links.

7. "web page"

In the Decision to Institute, we adopted Patent Owner's construction that a "web page" is "a document on the World Wide Web." Inst. Dec. 11. Patent Owner now contends that "[a] page corresponds to a textual object in a network, while a node is a representation of a textual object." PO Resp. 21–22 (citing Ex. 1001, 12:40–45). We evaluate whether the correct interpretation of "web page" is a document on the World Wide Web that corresponds to a textual object in a network, in contrast to a node, which is a representation of a textual object.

In the excerpt of the '571 Patent specification cited by the Patent Owner, a node is described as follows.

> A node 2008 is any entity that can be represented by a box on a display 38 such as a GUI 70. A node 2008 might be, for example, an object in a database 54, a portion of an object in a database 54, a document, a section of a document, a World Wide Web page, or an idea or concept, such as a topic name.

Ex. 1001, 12:40–45.

Because the specification of the '571 Patent states that a node might be a World Wide Web page (*id.*), we determine that the correct interpretation of "web page" should not distinguish a web page from a node.

Our construction of "web page" is consistent with the specification of the '571 Patent, which states, "[a] web page is usually a document." *Id.* at 48:30–31. Accordingly, we determine that our prior interpretation of "web page" as "a document on the World Wide Web" is correct.

8. Order of Steps

Patent Owner contends that Petitioner has not shown that Fox Thesis and Fox SMART teach the step of deriving actual cluster links, as recited in claim 21. PO Resp. 48–49. Patent Owner's contention is based, in part, on claim 21 reciting a specific order of steps. *Id*. In particular, claim 21 recites "generating a set of candidate cluster links for nodes indirectly related to the chosen node" and "deriving actual cluster links from the set of candidate cluster links." Patent Owner's contention is based on these steps being performed in the order outlined above.

Petitioner does not take a position on whether the elements of the challenged claims require the specific arrangement argued by Patent Owner. Petitioner contends that the asserted prior art teaches the steps as arranged in the challenged claims. *See e.g.*, Reply 8.

We determine that actual cluster links can be derived from candidate cluster links only if the candidate cluster links already have been generated. This determination is consistent with the specification of the '571 Patent. In particular, the specification of the '571 Patent explains that candidate cluster links are the set of all possible cluster links between a search node and a target node. Ex. 1001, 21:67–22:3. The '571 Patent specification continues that actual cluster links are a subset of the candidate cluster links, which meet certain criteria. *Id.* at 22:2–4.

We agree with Patent Owner that claim 21 recites a specific arrangement with respect to the two steps of generating candidate cluster links and deriving actual cluster links from the candidate cluster links. In particular, actual cluster links are derived from candidate cluster links after the candidate cluster links have been generated.

B. Alleged Obviousness of claims 12, 21, and 22 over Fox Thesis, Fox SMART, and Envision

Petitioner contends that claims 12, 21, and 22 of the ’571 Patent are unpatentable, under 35 U.S.C. § 103, as they would have been obvious over the combination of Fox Thesis, Fox SMART, and Envision. Pet. 9–21. In support of the asserted ground of unpatentability, Petitioner sets forth the teachings of the cited prior art, provides detailed claim charts, and cites to the declaration of Dr. Fox (Ex. 1003 ¶¶ 218–235), explaining how each limitation is taught in the cited prior art combination. Pet. 9–21.

The claim chart persuasively reads all elements of each of claims 12 and 22 onto the teachings of Fox Thesis, Fox SMART, and Envision, taken together. Despite the counter-arguments in Patent Owner’s Response, and the evidence cited therein, which we have also considered, Petitioner has shown by a preponderance of the evidence that claims 12 and 22 of the ’571 Patent are unpatentable, under 35 U.S.C. § 103, as they would have been obvious over the combination of Fox Thesis, Fox SMART, and Envision. Petitioner, however, has not shown by a preponderance of the evidence that claim 21 of the ’571 Patent is unpatentable, under 35 U.S.C. § 103, as obvious over the combination of Fox Thesis, Fox SMART, and Envision.

1. Fox Thesis

Fox Thesis describes improving query and document representation schemes for information retrieval. Ex. 1012, 261. In particular, useful types of bibliographic data are incorporated into a model to test clustering and retrieval functions. *Id.* at 164.

Bibliographic connections between articles are illustrated for an exemplary set “O” of documents, which are represented by letters A through

G. Ex. 1012, 165–66; Fig. 6.2. This exemplary set "O" includes direct and indirect citation references. *Id.* at 166–67; Table 6.2.

Based on the reference pattern for a set of documents, various measures of the interconnection between the documents may be derived. Ex. 1012, 166. For example, weights are assigned "based upon integer counts" for bibliographically coupled documents. *Id.* at 167.

Citation submatrices represent reference or citation information. Ex. 1012, 169–70. For example, submatrix <u>*bc*</u> represents bibliographically coupled reference information and submatrix <u>*cc*</u> represents co-citation reference information. *Id*. at 169–72; Figs. 6.3–6.5.

2. Fox SMART

The System for Mechanical Analysis and Retrieval of Text (SMART) is described as a project for designing a fully automatic document retrieval system and for testing new ideas in information science. Ex. 1013, 3. Fox SMART describes an implementation in which software components of SMART are implemented in the C Programming Language and run under the UNIX™ operating system on a VAX™ 11/780 computer. *Id.* at 1, 4.

In SMART, an automatic indexing component constructs stored representations of documents. Ex. 1013, 3. Bibliographic information is used to enhance document representations. *Id*. at 29. The SMART system may process basic raw data, such as an exemplary "N" collection of articles and citation data describing which articles are cited by others. *Id.*at 29–30. The exemplary input data includes indirect citation relationships, such as bibliographic coupled and co-citation relationships. *Id.* at 30–32.

A clustering algorithm is processed by the SMART system as follows: "[t]he clustering algorithm produces a hierarchy where all *N* documents in a

collection end up as leaves of a multilevel tree . . . Clustering proceeds by adding documents one by one starting with an initially empty tree." Ex. 1013, 44. Adding documents involves finding the proper place to insert, attaching the incoming entry appropriately, and recursively splitting overly large nodes. *Id*. at 47.

3. Envision

Envision describes integrating retrieval systems with the World Wide Web. Ex. 1006, 482. In particular, Envision describes the World Wide Web as a wide area hypertext system and indicates that hypertext and hypermedia linking must be coordinated with various approaches of search and retrieval. *Id*.

4. Claim 12

Petitioner's claim chart persuasively reads all elements of claim 12 onto the combined teachings of Fox Thesis, Fox SMART, and Envision. Pet. 11–16 (citing Ex. 1006, 482–84, 487, 482, Fig. 4; Ex. 1012, 164, 166–68, 176, 181, 186, 187, 193, 195, 199–200, 207–09, 213, 237–39; Ex. 1013, 13–15, 27, 29, 30–32, 36–38, 41, 44, 46, 47, 49, 50, 53–54; 1003 ¶¶ 159–164.) For instance, the combination of Fox Thesis, Fox SMART, and Envision teaches "identifying Universal Resource Locators for the web pages," as recited in claim 12. In particular, Fox SMART teaches citation data indicating which articles cite articles, point to cited articles, or point away from the referring article. Ex. 1013, 30 ("The set captures the details of which of the *N* collection articles are cited by others; the arrow in (4-1) points from referring to cited article.")

As an additional example, the combination of Fox Thesis, Fox SMART, and Envision teaches "cluster analyzing Universal Resource

Locators for indirect relationships," as recited in claim 12. Both Fox SMART (Ex. 1013, 44) and Fox Thesis (Ex. 1012, 193) teach a clustering algorithm that produces a hierarchy, or classification, in which all of the *N* documents in the collection end up as leaves of a multilevel tree. The clustering algorithm is performed using indirect relationships, including coupling and co-citation relationships. *See e.g.*, Ex. 1013, 46 ("the overall similarity between documents can be determined based on available subvectors . . . ."); *see also id*. at 31 ("Constructing $\overrightarrow{bc}$ Subvectors . . . . hav[ing] *n* units of coupling . . . .").

As to whether Petitioner has satisfied the requirements for combining the teachings of Fox Thesis, Fox SMART, and Envision, we determine that Petitioner has articulated sufficient reasoning with a rational underpinning as to why one of ordinary skill in the art would have combined the retrieval systems taught in Fox Thesis and Fox SMART with documents stored as web pages and linked by hypertext and hypermedia linking taught in Envision. *See KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007) (citing *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)). Petitioner's Declarant, Dr. Edward A. Fox, states that when the World Wide Web became available, it was obvious for researchers in the field of information retrieval to study and build systems using the World Wide Web. Ex. 1003 ¶ 223. We give Dr. Fox's statement substantial weight because it is consistent with the teaching in Envision of the emergence of the World Wide Web and coordinating hypertext and hypermedia linking with various approaches of search and retrieval. Ex. 1006, 482.

Additionally, Dr. Fox states it would have been obvious to one of ordinary skill in the art to combine the techniques of Envision, Fox Thesis,

and Fox SMART because Envision was built on, and was itself, a follow-on work to Fox Thesis and Fox SMART. Ex. 1003 ¶¶ 154–55, 224–30. We give Dr. Fox's statement substantial weight, because his statement is consistent with the teachings in Fox Thesis, Fox SMART, and Envision. *See* Ex. 1006, 482 ("users prefer vector and feedback methods"), 491 (citation to prior work of Dr. Fox including a text regarding integrating search and retrieval with hypertext); Ex. 1012, 343 (Fox Thesis cites to Fox SMART); Ex. 1013, 84 (Fox SMART cites to Fox Thesis).

We now address Patent Owner's counter-arguments in turn. Patent Owner, in reliance on its Declarant, Dr. Jacobs, contends that Fox Thesis, Fox SMART, and Envision, taken together do not teach "identifying Universal Resource Locators for the web pages" and "cluster analyzing the Universal Resource Locators for indirect relationships," as recited in claim 12. PO Resp. 14.

First, Patent Owner contends, "[t]he prior art in general is devoid of any teaching or suggestion of applying citation analysis to the web or to hypertext networks prior to 1996." PO Resp. 15–16 (citing Ex. 2113 ¶ 89). Patent Owner acknowledges "[t]he Web was known before 1996, and hypertext had been known long before the Web." *Id.* at 15. Patent Owner's Declarant, Dr. Jacobs, however, states, "[w]hile it may have been obvious to combine some information retrieval methods with the Web in 1996 . . . the inventive step of the '571 patent of treating Web links as citations was by all indications non-obvious." Ex. 2113 ¶ 89.

Envision, however, teaches applying citation analysis to hypertext systems, including the World Wide Web.

> We are beginning to see the emergence of *wide area hypertext systems* (Yankelovich, 1990) *like the WorldWideWeb (WWW)*,

> that carry this concept forward into a distributed environment. Clearly, *we must coordinate hypertext and hypermedia linking with the various approaches to search and retrieval* (Fox et al., 1991b). *One approach* is the idea of information graphs (including hypergraphs), where *objects of all types are interrelated by links or arcs that capture not only citation (reference)* but also inheritance, inclusion, association, synchronization, sequencing, *and other relationships*.

Ex. 1006, 482 (emphasis added).

Patent Owner, in reliance on Dr. Jacobs, states that the above-referenced excerpt of Envision is not sufficient because, "[n]owhere in the referenced section[] does it say that hyperlinks or hypertext would be treated as citations for purposes of analysis." PO Resp. 18 (citing Ex. 2113 ¶ 92). Patent Owner and Dr. Jacobs's statements are inaccurate representations of the reference. The approach taught in Envision is interrelating "objects of all types," including objects on the World Wide Web, so as to capture citation relationships (Ex. 1006, 482).

Nonetheless, Patent Owner addresses each of the references separately. PO Resp. 15–21. One cannot show non-obviousness by attacking references individually where the challenge is based on a combination of references. *In re Merck & Co., Inc.*, 800 F.2d 1091, 1097 (Fed. Cir. 1986); *In re Keller*, 642 F.2d 413, 426 (CCPA 1981). As discussed above, we find that Petitioner has articulated sufficient reasoning with a rational underpinning as to why one of ordinary skill in the art would have combined the teachings of Fox Thesis, Fox SMART, and Envision.

Second, Patent Owner contends that "the deficiencies of the cited art" are evident because Petitioner's Declarant, Dr. Fox, improperly conflates "page" with "node." PO Resp. 21. As discussed above with respect to claim

construction, we determine that the proper interpretation of "web page" should not distinguish a web page from a node.

Third, Patent Owner, relying on its Declarant, contends that "Fox SMART does not specifically teach using bc and cc in clustering, but suggests that some combination of terms with 'bibliographic connections' can be used." PO Resp. 24 (citing Ex. 2113 ¶ 194). Patent Owner's Declarant, however, provides conclusory statements that do not take into account all of the relevant teachings of Fox SMART. We find that Fox SMART teaches that the clustering algorithm is performed using indirect relationships based on the following teachings. First, Fox SMART teaches constructing bibliographic and co-citation subvectors. Ex. 1013, 30–32. Bibliographic and co-citation are examples of indirect relationships between documents. *Id*. Second, Fox SMART teaches determining a similarity between documents based on these subvectors. *Id*. at 46. Third, Fox SMART teaches that the determined similarity is used in the clustering process to form the tree. *Id*. at 49–50.

Patent Owner additionally contends that the combination of Fox Thesis, Fox SMART, and Envision does not teach "displaying identities of web pages." PO Resp. 27. We, however, find that Petitioner has shown by a preponderance of the evidence that the combination of Fox Thesis, Fox SMART, and Envision teaches displaying identities of web pages, as recited in claim 12. The SMART system presents documents in a retrieved cluster to the user. Ex. 1013, 54 ("most of the documents in a retrieved cluster are presented to the user . . . .") In addition, the Envision system displays search results. Ex. 1006, 487 ("Central to the search results display design is the concept of viewing each document (item) as a node within the Envision

database graph . . . . The search results design provides a graphical, direct manipulation presentation of documents found by the search . . . ."). Patent Owner contends that Petitioner fails to show displaying the results because Petitioner fails to show the claimed process. For the reasons given above, we are not persuaded.

Patent Owner's remaining contentions relate to whether the Petitioner has satisfied the requirements for combining the teachings of Fox Thesis, Fox SMART, and Envision. For example, Patent Owner contends that Petitioner improperly relies on hindsight bias. PO Resp. 28.

As indicated above, we determine that Petitioner has articulated sufficient reasoning with a rational underpinning as to why one of ordinary skill in the art would have combined the retrieval systems taught in Fox Thesis and Fox SMART with documents stored as web pages and linked by hypertext and hypermedia linking taught in Envision. *See KSR*, 550 U.S. at 398. For instance, in addition to the explicit suggestion in Envision (*see* Ex. 1006, 482), Dr. Fox wrote or was co-author of each of Fox Thesis, Fox SMART, and Envision. *See* Ex. 1006, 480; Ex. 1012, iii; Ex. 1013, 1.

Patent Owner further contends that processing web based links in the manner claimed would not have been predictable at the time of the invention of the '571 Patent. PO Resp. 30. In particular, Patent Owner contends that Google's introduction of its algorithms took experts in the field by surprise and was considered a major breakthrough. PO Resp. 30–31 (citing Ex. 2113 ¶ 177; Ex. 2114 ¶¶ 22, 51–63). Patent Owner's contention is based on its view that the combined teachings of Fox Thesis, Fox SMART, and Envision are not sufficient because they do not teach computerized searching of an electronic database. PO Resp. 32; *see also* Tr. 49:15–18 ("[T]he Fox papers

by themselves don't get you there . . . every one . . . is directed to printed articles, not an electronic database."). According to Patent Owner, the prior art cited by Petitioner teaches experiments that are not directed to web based documents, "but rather are directed toward limited experimentation with bibliographic relationships existing among paper documents." PO Resp. 1.

We disagree with Patent Owner. For example, Fox SMART teaches an implementation in which software components of SMART are implemented in the C Programming Language and run under the UNIX™ operating system on a VAX™ 11/780 computer. Ex. 1013, 1, 4. In SMART, an automatic indexing component constructs stored representations of documents. *Id*. at 3. In light of the various teachings of Fox Thesis, Fox SMART, and Envision discussed herein, we determine that Fox Thesis, Fox SMART, and Envision, taken together, teach computerized searching of an electronic database.

Patent Owner contends that Google's search engine using its PageRank algorithm is objective indicia of non-obviousness. PO Resp. 56–59. As an initial matter, Patent Owner's contentions again appear to be based on its view that the combined teachings of Fox Thesis, Fox SMART, and Envision are not sufficient because they do not teach computerized searching of an electronic database. *Id.* at 56 ("Link analysis technology applied to the Web, as claimed in the '571 patent and embodied in PageRank, satisfied a long felt need for improved *computerized search*." (Emphasis added (citation omitted).)); Tr. 60:24–63:2 ("[I]t certainly wouldn't have been obvious to one of ordinary skill based on Fox's work to extend these ideas from this paper collection to electronic databases."). For the reasons discussed above, we disagree with Patent Owner's view and

determine that Fox Thesis, Fox SMART, and Envision, taken together, teach computerized searching of an electronic database.

Furthermore, we note that Patent Owner has not shown that the asserted success of a commercial embodiment of the '571 patent actually resulted from features recited in the claims of the '571 patent. As explained in the Declaration of Dr. Jacobs, Google's "legendary insight was to rate pages based on the number and importance of links that pointed to them." Ex. 2113 ¶ 183 (citing Ex. 2045). Patent Owner refers to the patented technology of the '571 patent as solving a problem of distinguishing relevant search results from irrelevant results by analyzing non-semantical and indirect citation relationships. PO Resp. 57. For the reasons discussed above with respect to claim construction, Patent Owner has not explained persuasively or provided sufficient evidence to support limiting claim 12 to analysis involving non-semantical relationships. Nonetheless, Dr. Jacobs statement regarding Google's technology applies to direct citations. Also, Patent Owner's characterization of Google's technology indicates that "importance" of links is used to rate pages, which is too vague to be considered as persuasive evidence of a nexus between Google's technology and claim 12.

Patent Owner also points to Google's license of the '571 Patent. PO Resp. 56. Patent Owner, however, admits that this license resulted in the settlement of a lawsuit (*id.*), which without additional contextual evidence, weighs against finding a nexus.

Additionally, we determine that in light of the weak showing of secondary considerations, the evidence of obviousness with respect to Fox Thesis, Fox SMART, and Envision, is sufficient to support the conclusion

that claim 12 would have been obvious. *See Leapfrog Enterprises, Inc. v. Fisher-Price, Inc.*, 485 F.3d 1157, 1162 (Fed. Cir. 2007). As discussed above, Petitioner has provided a strong case of obviousness. For example, Petitioner has pointed to an explicit suggestion in Envision to combine the references (Ex. 1006, 482) and has provided declaration testimony of Dr. Fox (Ex. 1003 ¶¶ 154–55, 223–30), who wrote or was co-author of each of the references (Ex. 1006, 480; Ex. 1012, iii; Ex. 1013, 1).

Accordingly, even after considering the counter-arguments in Patent Owner's Response, and the evidence cited therein, we find that Petitioner has shown by a preponderance of the evidence that claim 12 is unpatentable as it would have been obvious over the combination of Fox Thesis, Fox SMART, and Envision.

5. Claim 21

Petitioner points to teachings in Fox Thesis and Fox SMART of computing a similarity to be used in a clustering algorithm to satisfy the following element of claim 21, "determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node using the identified hyperjump data, wherein the step of determining comprises non-semantically generating a set of candidate cluster links for nodes indirectly related to the chosen node using the hyperjump data, assigning weights to the candidate cluster links and deriving actual cluster links from the set of candidate cluster links based on the assigned weights." Pet. 17–18 (citing Ex. 1012, 166–68, 170–72, 174–77, 181, 195, 272; Ex. 1013, 30–32, 36, 46; Ex. 1003 ¶¶ 161–63, 172–174). In particular, according to Petitioner's Declarant, a tree resulting from the clustering algorithm, which includes "all" the documents in the collection teaches the

claimed set of generated candidate cluster links. Ex. 1003 ¶ 160 ("Fox SMART . . . disclose[s] **generating candidate cluster links** . . . [f]or example, the particular clustering analysis that I employed builds a tree.").

Petitioner, however, points to this same clustering algorithm for deriving actual cluster links from the candidate cluster links, as recited in claim 1. Pet. 17–18 (citing Ex. 1003 ¶ 162). In particular, Petitioner's Declarant points to Fox SMART's teaching of concentration tests performed as part of the clustering algorithm. Ex. 1003 ¶ 162 (citing Ex. 1013, 49–51). As taught in Fox SMART:

> Candidate clusters which pass the concentration test are those formed by having enough highly correlated pairs in the proposed cluster. . . .
>
> "Uncour" repeatedly considers the remaining cluster that is most heavily covered by other clusters. If the overlap is too much, it is deleted. Eventually only clusters that pass all appropriate tests are accepted.

Ex. 1013, 50–51.

Petitioner's Declarant, Dr. Fox, testifies that Fox SMART derives a subset because "clusters that do not pass all the concentration and overlap tests are deleted." Ex. 1003 ¶ 162 (citing Ex. 1013, 51). Dr. Fox supplements his testimony by stating, "Fox SMART teaches that potential (i.e., candidate) clusters are rejected ('deleted') if they fail any one of these tests." Ex. 1032 ¶ 281 (citing Ex. 1005, 51). Dr. Fox further states that claim 21 does not require that candidate cluster links be deleted. *Id*.

Patent Owner contends that Dr. Fox incorrectly states that clusters that do not pass all of the tests are deleted. PO Resp. 33 (citing Ex. 2113 ¶¶ 82–103). Patent Owner's Declarant, Dr. Jacobs, provides his analysis of various aspects of the clustering process and concludes that the clustering algorithm,

including the concentration tests noted above, does not result in a subset. Ex. 2113 ¶¶ 82–103. In particular, Dr. Jacobs states that the clustering process involves accepting trial clusters, which then must pass the concentration tests to become candidate clusters. *Id*. ¶ 91. Dr. Jacobs also states that clusters are deleted only in the case of overlap with a new group of clusters formed from splitting. *Id*. ¶ 88. Additionally, Dr. Jacobs states that moving orphans to the garbage cluster does not result in deleting those orphans or creating a subset. *Id*. ¶ 101. Dr. Jacobs, instead, states that the orphans may not be garbage in the end as they may be assigned a node as new documents are added to the tree. *Id*.

We determine that the statements of Patent Owner's Declarant are consistent with Fox SMART's teaching of clustering. Fox SMART teaches that the clustering process initializes a new tree as empty, adds documents to the tree, and recursively splits overly large nodes of the tree. Ex. 1013, 47. Fox SMART states that splitting is accomplished by the following procedures div_cent, cleave, and uncour. *Id*. at 49.

Fox SMART further states:

> First a complete similarity matrix is formed based on the pairwise combined similarity values. "Cleave" then identifies a plausible clustering except that no limit on overlap is considered. "Uncour" compensates for that by first deleting clusters that exhibit too much overlap with remaining clusters, and secondly by assigning the others to a "garbage" or "orphan" cluster.

Id.

As taught in Fox SMART, the concentration tests that are cited by Petitioner are performed as part of forming the cluster tree. *Id*. Petitioner does not identify a teaching in Fox SMART of deleting clusters other than

those that simply overlap, or duplicate, other clusters. Overlapping clusters are deleted following a routine that identifies plausible clustering with "no limit on overlap." *Id*. Additionally, Fox SMART includes code that collects orphans. *Id*. at 52. Petitioner, however, has not shown that Fox SMART teaches a subset, which does not include these orphans.

Dr. Fox's supplemental testimony that "Fox SMART teaches that potential (i.e., candidate) clusters are rejected ('deleted') if they fail any one of these tests" (Ex. 1032 ¶ 281 (citing Ex. 1005, 51)) suggests that the terms "rejected" and "deleted" are the same. We do not agree. In view of Dr. Fox's testimony in both of his Declarations and the testimony of Dr. Jacobs, we find that one of ordinary skill in the art reasonably would have understood Fox SMART as teaching deleting overlap that had been generated by the immediately preceding software routine. Additionally, we find that one of ordinary skill in the art reasonably would have understood Fox SMART as teaching that the tests referred to by Dr. Fox are processed during formation of the tree. We, therefore, determine that Dr. Fox's testimony does not address persuasively the requirement in claim 21 of deriving a subset of the already generated candidate cluster links.

In light of the Declaration by Patent Owner's Declarant, Dr. Jacobs, we determine that Petitioner has not shown by a preponderance of the evidence that the combination of Fox Thesis, Fox SMART, and Envision teaches deriving actual cluster links. For the foregoing reasons, Petitioner has not established that claim 21 of the '571 Patent is unpatentable, under 35 U.S.C. § 103, as obvious over the combination of Fox Thesis, Fox SMART, and Envision.

6. Claim 22

Petitioner's claim chart persuasively reads all elements of claim 22 onto the teachings of Fox Thesis, Fox SMART, and Envision, taken together. Pet. 11–21 (citing Ex. 1006, 482–84, 487, 482, Fig. 4; Ex. 1012, 164, 166–68, 170–71, 176, 181, 186, 187, 193, 195, 199–200, 207–09, 213, 237–39; Ex. 1013, 13–15, 26–27, 29–32, 35–38, 41, 44, 46, 47, 49, 50, 53–54; 1003 ¶¶ 159–164.) For instance, we determine that Petitioner has shown by a preponderance of the evidence that the combination of Fox Thesis, Fox SMART, and Envision teaches hyperjump data that has a direct reference and an indirect reference to a chosen node. For example, Fox Thesis teaches an exemplary set "O" of documents represented by letters A through G, which includes direct and indirect citation references. Ex. 1012, 165–67; Table 6.2. Fox SMART teaches data comprising a set of tuples that captures which of the articles in a collection are cited by other articles. Ex. 1013, 29–32. Fox SMART further teaches that these data include direct and indirect citations or references. *Id*. at 30–32. Additionally, as discussed above with respect to claim 12, Fox SMART teaches constructing bibliographic and co-citation subvectors. *Id*. Furthermore, according to the teachings of Envision, retrieval technologies should be coordinated with World Wide Web technologies, including hypertext and hypermedia linking. Ex. 1006, 482.

We also determine that Petitioner has shown by a preponderance of the evidence that the combination of Fox Thesis, Fox SMART, and Envision teaches determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node by proximity indexing the identified hyperjump data, as recited in claim 22. As discussed above

with respect to claim construction, "proximity indexing" means preparing data in a database for subsequent searching by organizing and categorizing the data based on their degree of relatedness to one another. Additionally, as discussed above, an example of proximity indexing is generating a set of candidate cluster links.

Both Fox SMART (Ex. 1013, 44) and Fox Thesis (Ex. 1012, 193) teach a clustering algorithm that produces a hierarchy, or classification, in which all of the *N* documents in the collection end up as leaves of a multilevel tree. We also find that Fox SMART teaches that the clustering algorithm is performed using indirect relationships because Fox SMART teaches determining a similarity between documents based on the biobligraphic and co-citation subvectors (Ex. 1013, 46) and using that determined similarity in the clustering process to form the tree (*id*. at 49–50).

We further determine that Petitioner has also shown by a preponderance of the evidence displaying one or more determined hyperjump data. For example, the SMART system presents documents in a retrieved cluster to the user. Ex. 1013, 54 ("most of the documents in a retrieved cluster are presented to the user."). Additionally, the Envision system displays search results. Ex. 1006, 487 ("Central to the search results display design is the concept of viewing each document (item) as a node within the Envision database graph. . . . The search results design provides a graphical, direct manipulation presentation of documents found by the search . . . ."). Furthermore, regarding recitation of "generating a source map" in claim 22, Envision teaches a graphic view window that allows each

document to be viewed as a node within the Envision database graph. Ex. 1006, 487.

Additionally, for the reasons discussed above with respect to claim 12, we determine that Petitioner has satisfied the requirements for combining the teachings of Fox Thesis, Fox SMART, and Envision.

Patent Owner argues that Petitioner has not made its showing for substantially similar reasons that Patent Owner provided for claim 12. PO Resp. 50–51. For the reasons discussed above with respect to claim 12, we disagree.

Patent Owner also contends that the cited art does not teach or suggest proximity indexing the identified hyperjump data. PO Resp. 51. Patent Owner relies on its Declarant for stating that Fox SMART cannot be considered to teach identifying direct reference data for nodes in a network. Pet. 52 (citing Ex. 2113 ¶ 236). Petitioner's Declarant, however, does not discuss Fox SMART's teaching of data comprising a set of tuples that captures which of the articles in a collection are cited by other articles (Ex. 1013, 29–32) including direct citations, in which k=1 (*id*. at 30) and indirect citations having *n* units of coupling (*id*. at 31).

For the foregoing reasons, Petitioner has shown by a preponderance of the evidence that claim 22 of the '571 Patent is unpatentable under 35 U.S.C. § 103(a) as it would have been obvious over Fox Thesis, Fox SMART, and Envision.

C. Motion to Exclude

Patent Owner filed a Motion to Exclude (Paper 44) in which Patent Owner seeks to exclude the Reply Declaration of Dr. Edward A. Fox (Ex. 1032) ("Reply Fox Declaration") submitted with Petitioner's Reply. Patent

Owner contends that the Reply Fox Declaration should be excluded in accordance with Federal Rule of Evidence ("FRE") 702. Paper 44, 1. In particular, Patent Owner contends that the Reply Fox Declaration should be excluded because FRE 702 precludes admission of expert testimony unless "the testimony is based on sufficient facts or data, the testimony is the product of reliable principles and methods, and the expert has reliably applied the principles and methods to the facts of the case." *Id*. With regard to this objection, we note that under 37 C.F.R. § 42.65(a), "[e]xpert testimony that does not disclose the underlying facts or data on which the opinion is based is entitled to little or no weight." Consequently, this objection properly goes to the weight to be given to Dr. Fox's testimony, and not to its admissibility.

It is within our discretion to assign the appropriate weight to the testimony offered by Dr. Fox. *See, e.g.*, *Yorkey v. Diab*, 601 F.3d 1279, 1284 (Fed. Cir. 2010) (holding the Board has discretion to give more weight to one item of evidence over another "unless no reasonable trier of fact could have done so"); *In re Am. Acad. of Sci. Tech Ctr.*, 367 F.3d 1359, 1368 (Fed. Cir. 2004) ("[T]he Board is entitled to weigh the declarations and conclude that the lack of factual corroboration warrants discounting the opinions expressed in the declarations."). When weighing evidence, we are capable of determining whether the prior art references render obvious the challenged claims without being confused, misled or prejudiced by Dr. Fox's testimony.

Patent Owner also contends that Dr. Fox has engaged in a pattern of giving false testimony. Paper 44, 1. For example, Patent Owner argues that that Dr. Fox's statements are false because Fox Thesis and Fox SMART

discuss paper documents, not an electronic database. *Id*. at 12. For the reasons discussed above with respect unpatentability, we do not agree.

Patent Owner additionally contends that Dr. Fox's opinions are based on incorrect constructions. *See e.g.*, Paper 44, 13. However, the construction of claim terms in a Decision to Institute is not final, and is reviewable in light of both parties' subsequent briefings and oral arguments. Inst. Dec. 9–10, 29.

For the reasons given, we deny Patent Owner's Motion to Exclude.

D. Motions to Seal

Patent Owner filed a Motion to Seal (Paper 30) the Declaration of Dr. Amy N. Langville ("Langville Declaration") filed as Exhibit 2114. Petitioner filed a Motion to Seal (Paper 37) the Transcript of the Deposition of Amy N. Langville, Ph.D. ("Langville Transcript") filed as Exhibit 1033. Both of these motions are unopposed.

Regarding Patent Owner's Motion to Seal, according to Patent Owner paragraphs 25, 112, and 113 of the Langville Declaration makes reference to certain facts about confidential licenses to the patents under review. Paper 30, 3. Additionally, Patent Owner contends that this information has not been made, and will not be made, public. *Id*.

Regarding Petitioner's Motion to Seal, according to Petitioner, Patent Owner has designated the transcript as confidential. Paper 37, 3. To avoid public disclosure, therefore, Petitioner submits sealing the Langville Transcript is appropriate. *Id*.

There is a strong public policy in favor of making information filed in *inter partes* review proceedings open to the public. *See Garmin Int'l v. Cuozzo Speed Techs., LLC*, Case IPR2012-00001 (PTAB March 14, 2013)

(Paper 34). Under 35 U.S.C. § 316(a)(1), the default rule is that all papers filed in an *inter partes* review are open and available for access by the public.[2] The standard for granting a motion to seal is "good cause." 37 C.F.R. § 42.54. A moving party bears the burden of showing that the relief requested should be granted. 37 C.F.R. § 42.20(c).

Regarding Patent Owner's Motion to Seal, Patent Owner, as the moving party, has failed to carry its burden. Patent Owner identifies only three paragraphs in the Langville Declaration that purportedly contain confidential information. However, Patent Owner has not pointed to proof in the record that any information contained in these paragraphs is confidential. Additionally, although Patent Owner contends that this information has not been made, and will not be made, public, Patent Owner presented this information during the hearing on October 30, 2014, which was open to the public. *See* Tr. 54:12–25. We, therefore, determine that Patent Owner has not met its burden of proof.

Regarding Petitioner's Motion to Seal, Patent Owner's designation of the transcript as confidential is not sufficient to show that the transcript contains confidential information. We, therefore, determine that Petitioner has not met its burden of proof.

We recognize a denial of the motions to seal would immediately unseal the material that Patent Owner desires to remain confidential and the effect would be irreversible. Therefore, rather than denying the motions at

[2] Additionally, we note that confidential information subject to a protective order ordinarily would become public 45 days after final judgment in a trial. Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,761 (Aug. 14, 2012). However, after denial of a petition to institute a trial or after final judgment in a trial, a party may file a motion to expunge confidential information from the record. 37 C.F.R. § 42.56.

this time, we will provide Patent Owner and Petitioner one week to (1) withdraw the motions to seal and request that we expunge Exhibits 2114 and 1033, or (2) withdraw the motions to seal, request that we expunge Exhibits 2114 and 1033, and replace them with redacted versions that leave out the confidential information.

We note that we have not relied on the three paragraphs of the Langville Declaration that Patent Owner identifies as containing allegedly confidential information. Other than this section discussing the Motion to Seal, our only mention of the Langville Declaration is a citation to the Patent Owner Response that includes a page number cite to the Langville Declaration.

III. CONCLUSION

We conclude that Petitioner has shown by a preponderance of the evidence that claims 12 and 22 of the ’571 Patent are unpatentable under 35 U.S.C. § 103, as they would have been obvious over Fox Thesis, Fox SMART, and Envision, taken together. We conclude that Petitioner has not shown that claim 21 of the ’571 Patent is unpatentable. Claims 26, 28, and 31 are not at issue in this trial.[3]

This is a final written decision of the Board under 35 U.S.C. § 318(a). Parties to the proceeding seeking judicial review of this decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

---

[3] In the Decision to Institute, we declined to institute an *inter partes* review of claims 26, 28, and 31 because we were not persuaded that Petitioner had shown that there was a reasonable likelihood of prevailing on its challenges to these claims. Inst. Dec. 2, 30.

IV. ORDER

For the reasons given, it is

ORDERED that claims 12 and 22 of U.S. Patent No. 6,233,571 are determined by a preponderance of the evidence to be unpatentable;

FURTHER ORDERED that claim 21 of U.S. Patent No. 6,233,571 is not determined to be unpatentable;

FURTHER ORDERED that Patent Owner's Motion to Exclude the Reply Declaration of Dr. Edward A. Fox (Exhibit 1032) is DENIED;

FURTHER ORDERED that Exhibit 2114 and Exhibit 1033 will be made available to the public after 5 PM Eastern five business days after the entry date of this decision, unless prior to that time, each of Patent Owner and Petitioner (1) withdraws the motions to seal and requests that we expunge Exhibits 2114 and 1033, or (2) withdraws the motions to seal, requests that we expunge Exhibits 2114 and 1033, and replaces them with redacted versions that leave out the confidential information; and

FURTHER ORDERED that, because this is a final written decision, parties to the proceeding seeking judicial review of the decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

FOR PETITIONER:

Heidi L. Keefe
COOLEY, LLP
ATTN: Heidi L. Keefe, Patent Docketing
777 6th Street N.W., Suite 1100
Washington, DC 20001
hkeefe@cooley.com
dcpatentdocketing@cooley.com.

David Silbert
KEKER & VAN NEST, LLP
633 Battery Street
San Francisco, CA 94111
djs@kvn.com
efiling@kvn.com

FOR PATENT OWNER:

Martin M. Zoltick
Nancy J. Linck
Soumya P. Panda
ROTHWELL, FIGG, ERNST & MANBECK, P.C.
607 14th St., N.W., Suite 800
Washington, DC 20005
mzoltick@rfem.com
nlinck@rfem.com
SRA-IPR@rfem.com

ADDENDUM 2

U.S. Patent No. 6,233,571

JA05000 – JA05092

2015-1652, 2015-1653

SOFTWARE RIGHTS ARCHIVE, LLC
v.
FACEBOOK, INC., LINKEDIN CORPORATION, TWITTER, INC.

US006233571B1

(12) United States Patent
Egger et al.

(10) Patent No.: US 6,233,571 B1
(45) Date of Patent: May 15, 2001

(54) **METHOD AND APPARATUS FOR INDEXING, SEARCHING AND DISPLAYING DATA**

(75) Inventors: **Daniel Egger**, 2027 W. Club Blvd., Durham, NC (US) 27705; **Shawn Cannon**, Hillsborough; **Ronald D. Sauers**, Mebane, both of NC (US)

(73) Assignee: **Daniel Egger**, Peachtree City, GA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/071,120**

(22) Filed: **May 4, 1998**

Related U.S. Application Data

(60) Division of application No. 08/649,304, filed on May 17, 1996, now Pat. No. 5,832,494, which is a continuation-in-part of application No. 08/076,658, filed on Jun. 14, 1993, now Pat. No. 5,544,352.

(51) **Int. Cl.**[7] **G06F 7/00**
(52) **U.S. Cl.** **707/2**; 707/3; 707/4; 707/5; 707/530
(58) **Field of Search** 707/2–5, 530

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,265,065		11/1993	Turtle .	
5,341,293	*	8/1994	Vertelney et al.	709/205
5,446,842	*	8/1995	Schaeffer et al.	709/205
5,530,852	*	6/1996	Meske, Jr. et al.	707/102
5,542,024	*	7/1996	Balint et al.	345/356

(List continued on next page.)

OTHER PUBLICATIONS

a) Agosti, et al., "A Two–Level Hypertext Retrieval Model for Legal Data," SIGIR '91 (1991).
b) Fowler, et al., "Integrating Query, Thesaurus and Documents Through a Common Visual Representation," SIGIR '91 (1991).
c) Rose & Belew, "Legal Information Retrieval: A Hybrid Approach," ICAIL '89 (1989).
d) Belew, Richard, "A Connectionist Approach to Conceptual Information Retrieval," ICAIL '87 (1987).

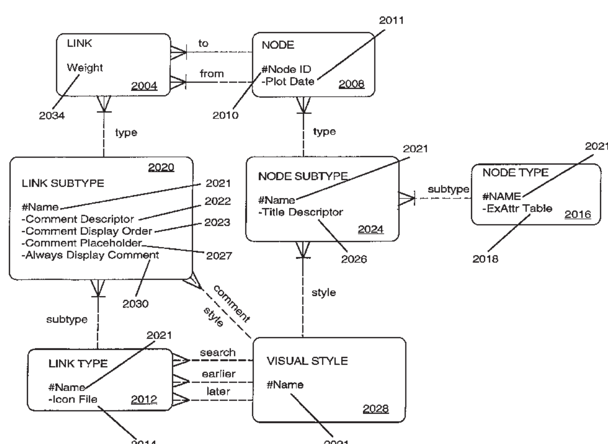
(List continued on next page.)

Primary Examiner—Wayne Amsbury
(74) *Attorney, Agent, or Firm*—Dorsey & Whitney LLP

(57) **ABSTRACT**

A computer research tool for indexing, searching and displaying data is disclosed. Specifically, a computer research tool for performing computerized research of data including textual objects in a database or a network and for providing a user interface that significantly enhances data presentation is described. Textual objects and other data in a database or network is indexed by creating a numerical representation of the data. The indexing technique called proximity indexing generates a quick-reference of the relations, patterns and similarity found among the data in the database. Proximity indexing indexes the data by using statistical techniques and empirically developed algorithms. Using this proximity index, an efficient search for pools of data having a particular relation, pattern or characteristic can be effectuated. The Computer Search program, called the Computer Search Program for Data represented in Matrices (CSPDM), provides efficient computer search methods. The CSPDM rank orders data in accordance with the data's relationship to time, a paradigm datum, or any similar reference. An alternative embodiment of the invention employs a cluster link generation algorithm which uses links and nodes to index and search a database or network. The algorithm searches for direct and indirect links to a search node and retrieves the nodes which are most closely related to the search node. The user interface program, called the Graphical User Interface (GUI), provides a user friendly method of interacting with the CSPDM program and prepares and presents a visual graphical display. The graphical display provides the user with a two or three dimensional spatial orientation of the data.

22 Claims, 56 Drawing Sheets



U.S. PATENT DOCUMENTS

5,544,352	*	8/1996	Egger	707/2
5,617,565	*	4/1997	Augenbaum et al.	707/102
5,649,186	*	7/1997	Ferguson	707/102
5,749,785	*	5/1998	Rossides	463/25
5,794,001	*	8/1998	Makone et al.	345/342
5,832,494	*	11/1998	Egger et al.	707/102
5,898,434	*	4/1999	Small et al.	345/348
6,098,081	*	8/2000	Heidorn et al.	707/501

OTHER PUBLICATIONS

e) Gelbart & Smith, “Beyond Boolean Search: FLEXICON, A Legal Text–Based Intelligent System,” ICAIL ’91 (1991).
f) Lin, “A Self–Organizing Semantic Map for Information Retrieval,” SIGIR ’91 (1991).
g) Turtle & Croft, “Inference Networks for Document Retrieval,” SIGR ’90 (1990).

* cited by examiner

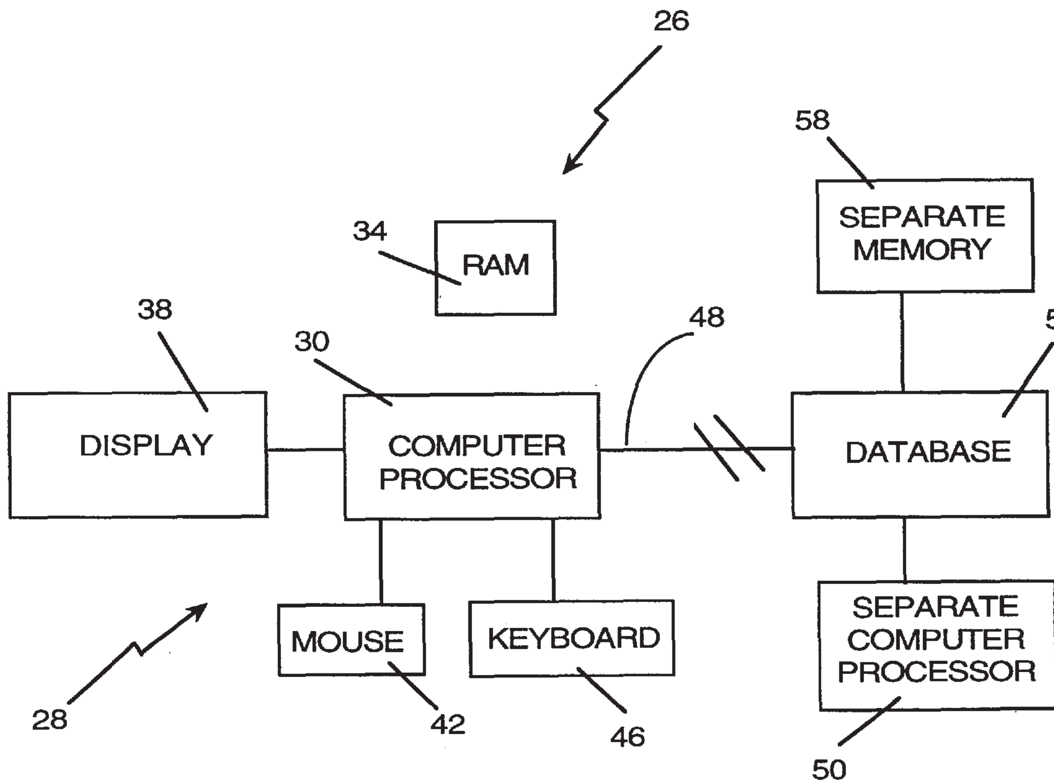

26
58
SEPARATE MEMORY
34
RAM
38
30
48
54
DISPLAY
COMPUTER PROCESSOR
DATABASE
MOUSE
KEYBOARD
SEPARATE COMPUTER PROCESSOR
28
42
46
50

Fig. 1

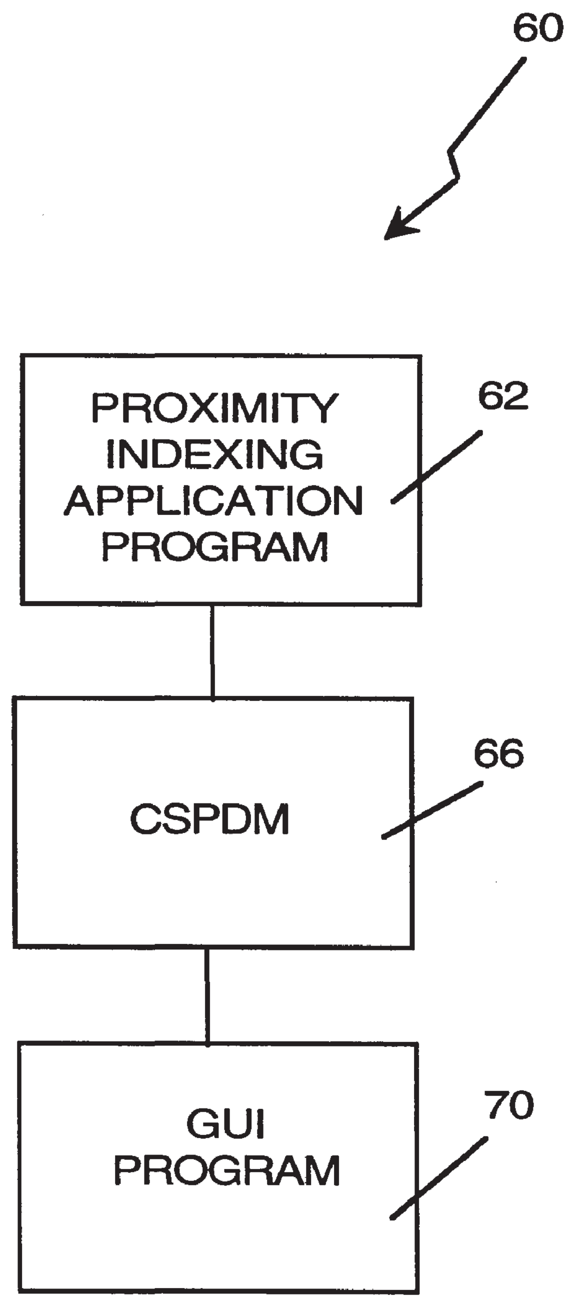

60
PROXIMITY INDEXING APPLICATION PROGRAM
62
CSPDM
66
GUI PROGRAM
70

Fig. 2

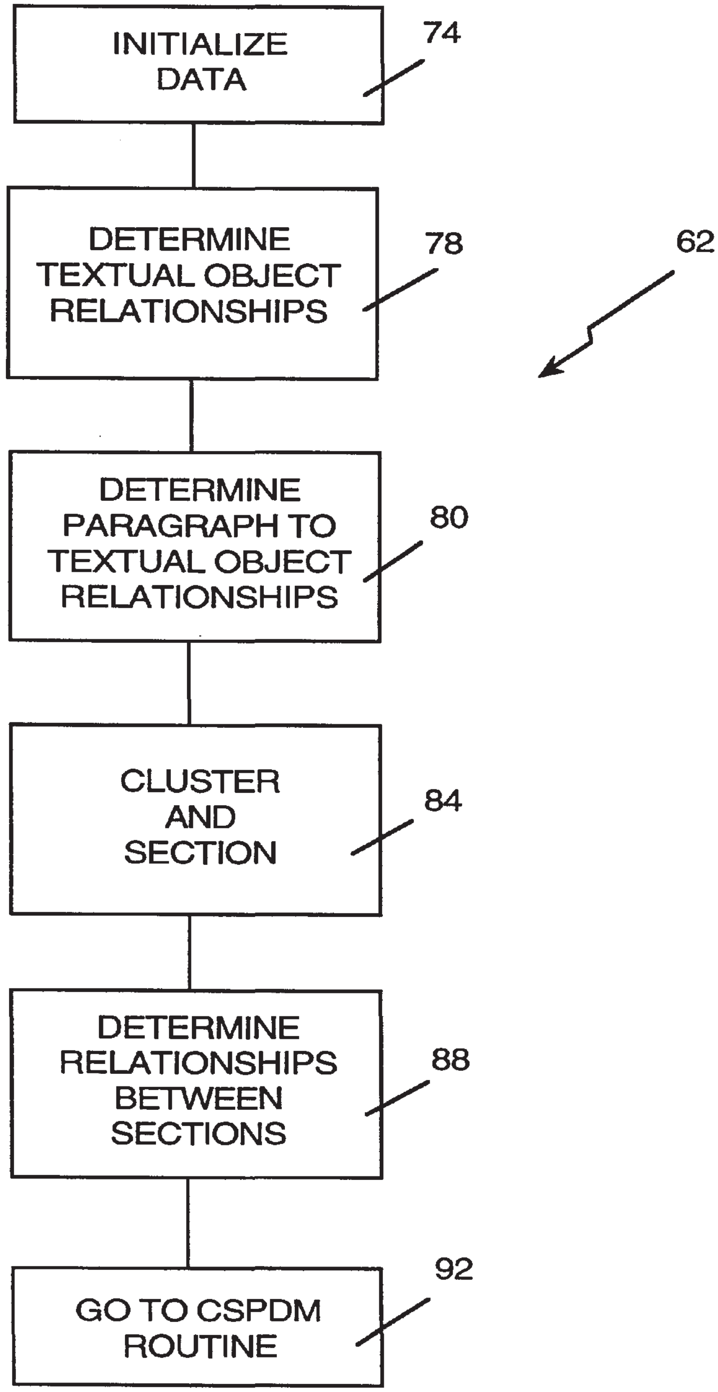
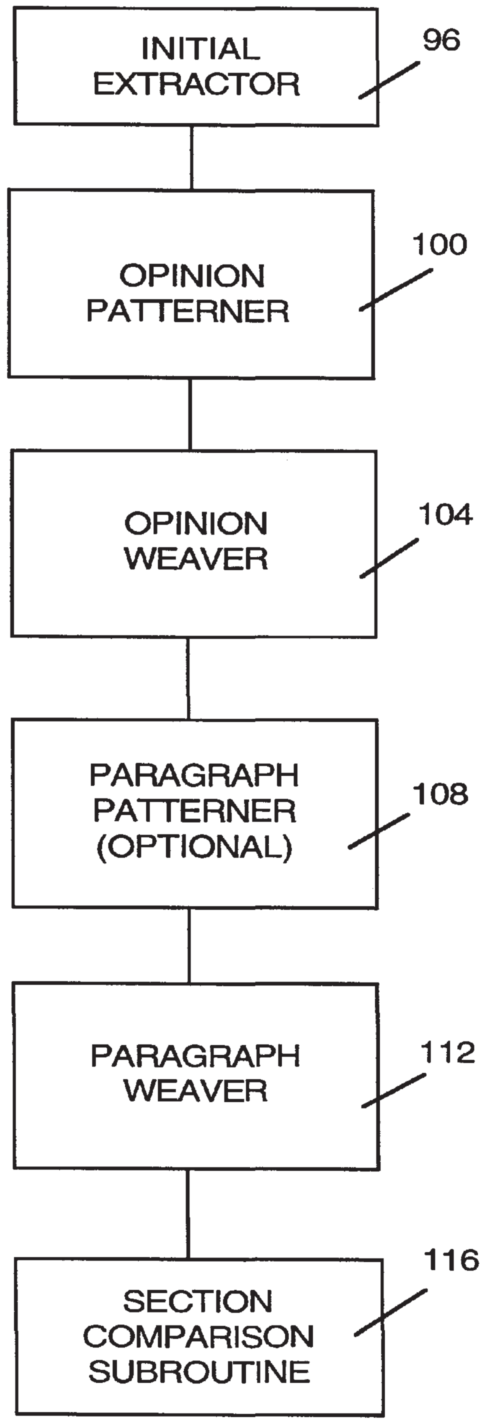

INITIALIZE DATA
74
DETERMINE TEXTUAL OBJECT RELATIONSHIPS
78
62
DETERMINE PARAGRAPH TO TEXTUAL OBJECT RELATIONSHIPS
80
CLUSTER AND SECTION
84
DETERMINE RELATIONSHIPS BETWEEN SECTIONS
88
GO TO CSPDM ROUTINE
92

Fig. 3A



INITIAL EXTRACTOR
96
OPINION PATTERNER
100
OPINION WEAVER
104
PARAGRAPH PATTERNER (OPTIONAL)
108
PARAGRAPH WEAVER
112
SECTION COMPARISON SUBROUTINE
116

Fig. 3B

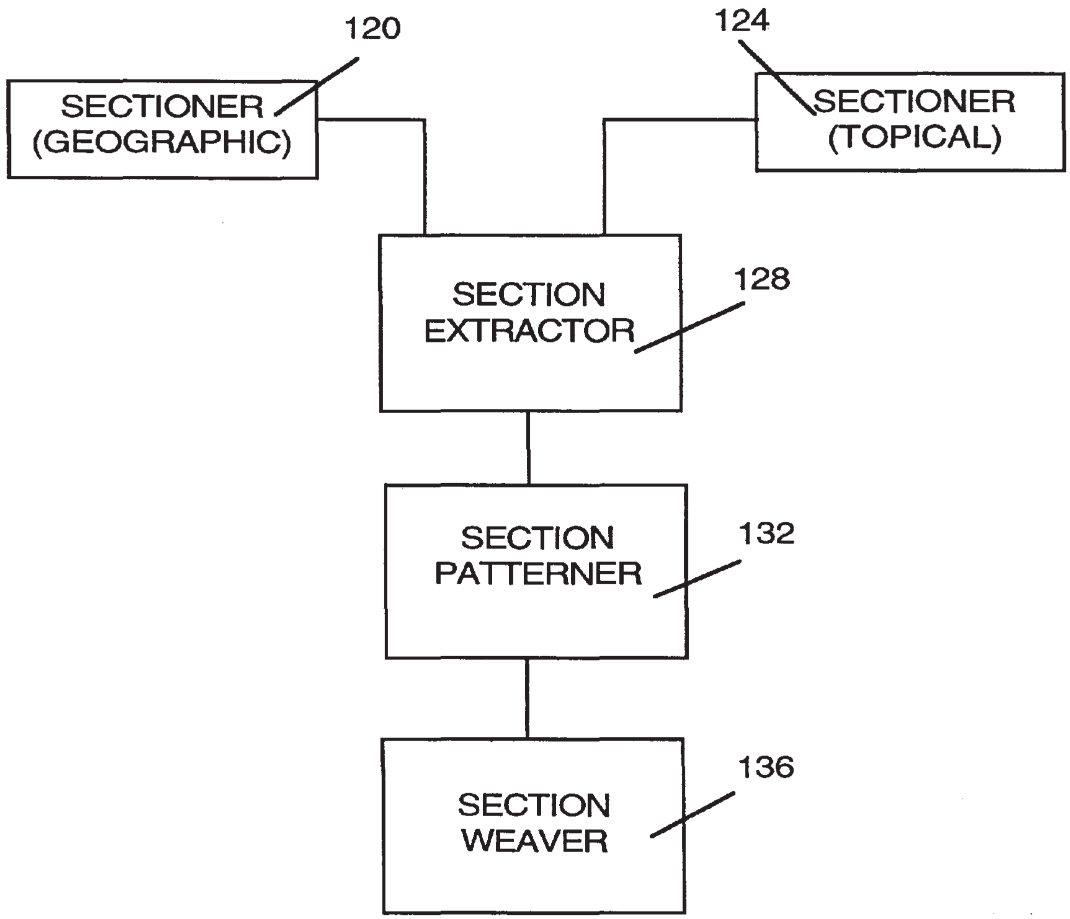

120
SECTIONER (GEOGRAPHIC)
124
SECTIONER (TOPICAL)
SECTION EXTRACTOR
128
SECTION PATTERNER
132
SECTION WEAVER
136

Fig. 3C

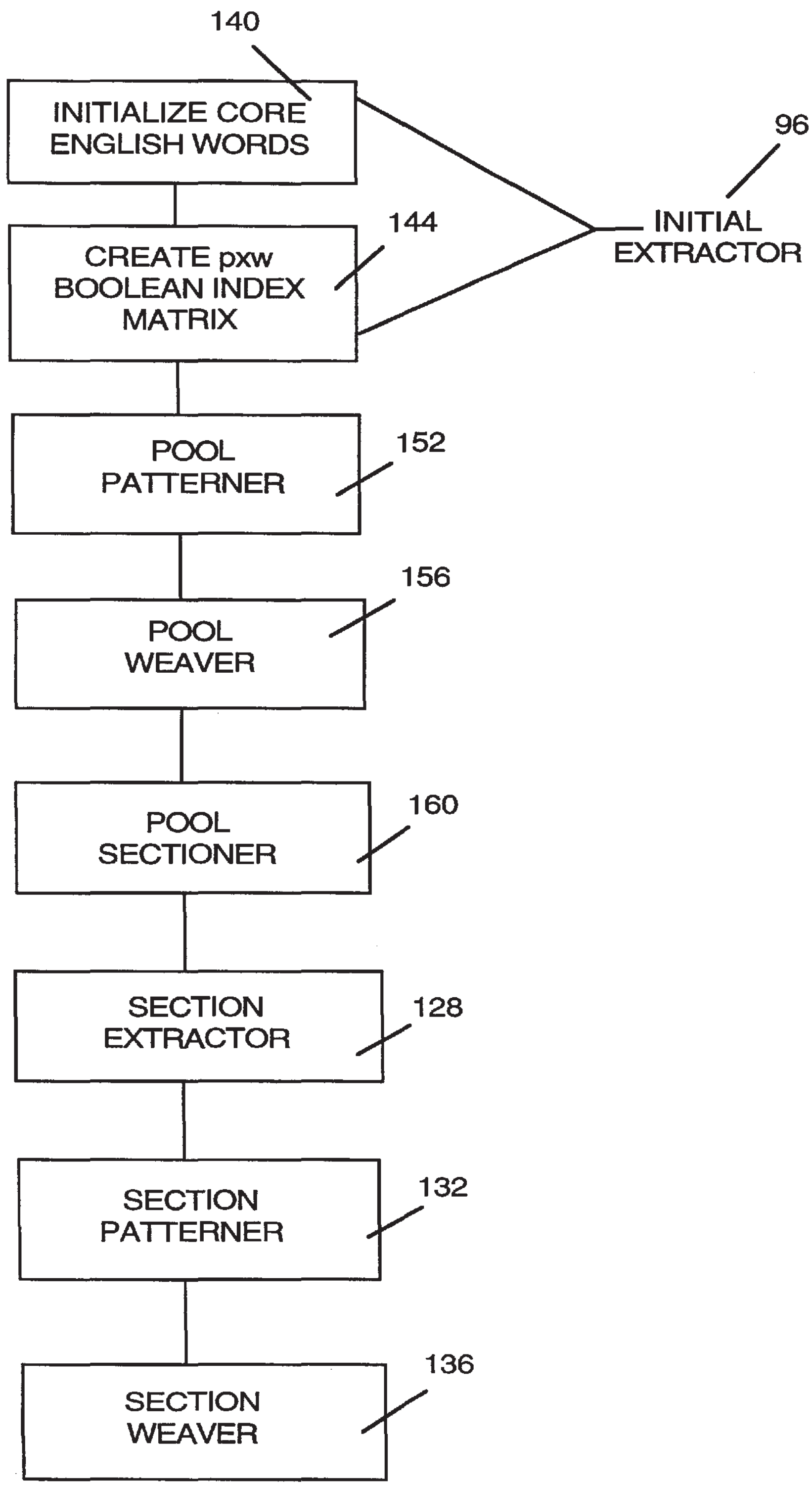
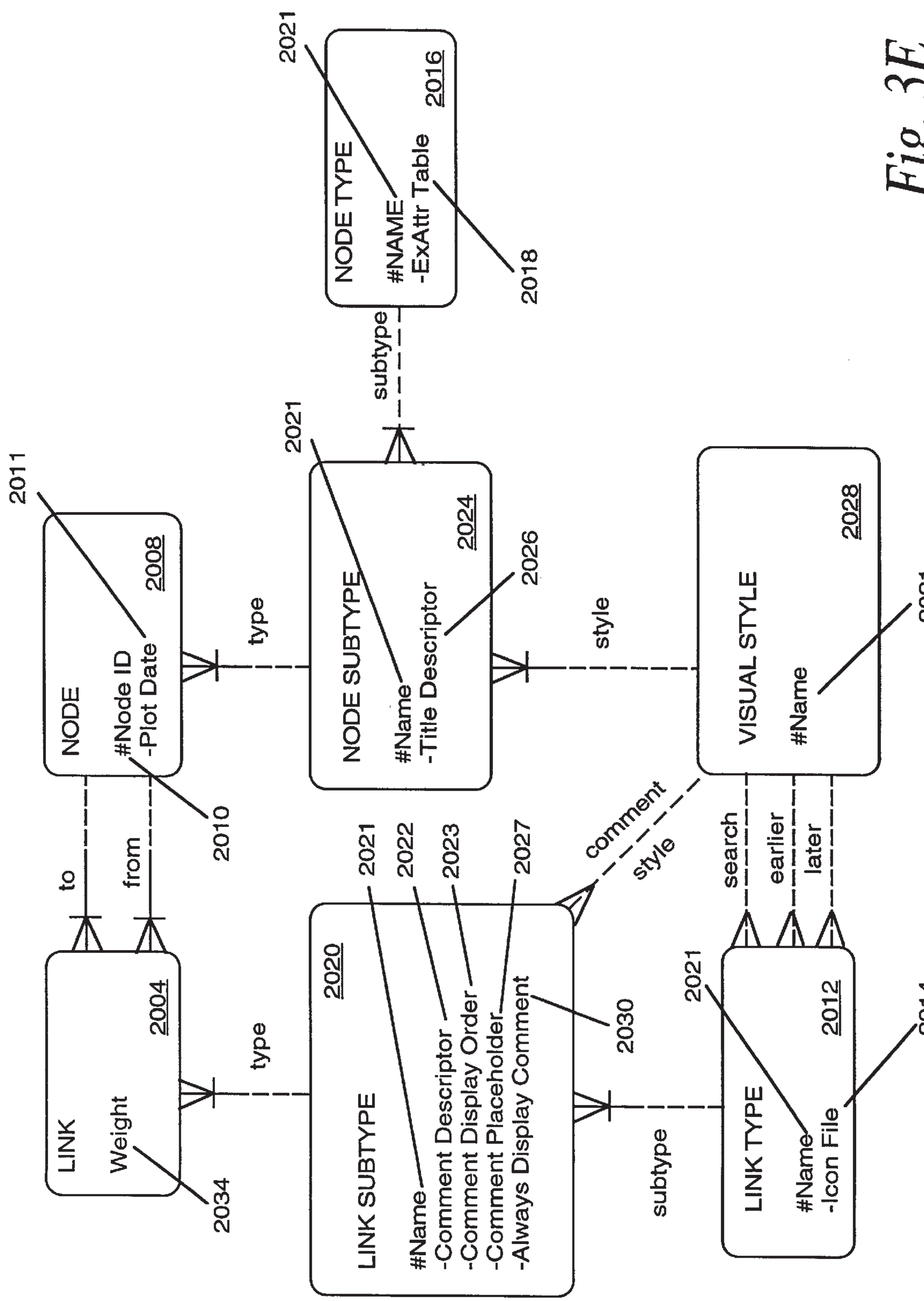

140
INITIALIZE CORE ENGLISH WORDS
96
INITIAL EXTRACTOR
144
CREATE pxw BOOLEAN INDEX MATRIX
POOL PATTERNER
152
POOL WEAVER
156
POOL SECTIONER
160
SECTION EXTRACTOR
128
SECTION PATTERNER
132
SECTION WEAVER
136

Fig. 3D



NODE TYPE
#NAME
-ExAttr Table
2016
2021
2018
subtype
NODE
#Node ID
-Plot Date
2008
2011
2010
type
NODE SUBTYPE
#Name
-Title Descriptor
2024
2021
2026
style
VISUAL STYLE
#Name
2028
2021
to
from
LINK
Weight
2004
2034
type
LINK SUBTYPE
#Name
-Comment Descriptor
-Comment Display Order
-Comment Placeholder
-Always Display Comment
2020
2021
2022
2023
2027
2030
comment
style
search
earlier
later
subtype
LINK TYPE
#Name
-Icon File
2012
2021
2014

Fig. 3E

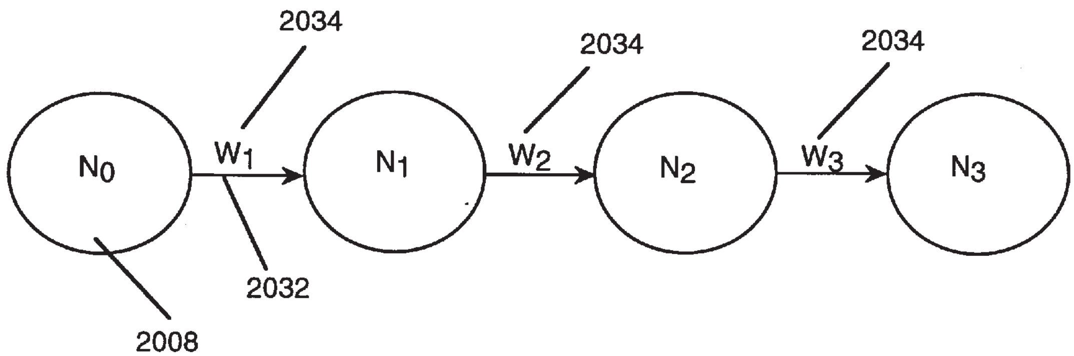

2034
2034
2034
N0
W1
N1
W2
N2
W3
N3
2032
2008

Fig. 3F

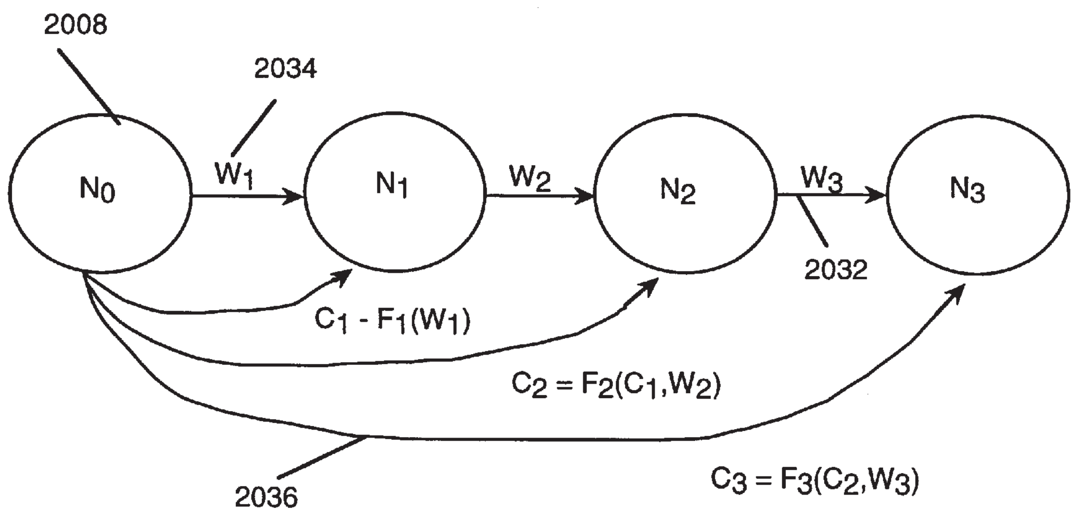

2008
2034
N0
W1
N1
W2
N2
W3
N3
2032
C1 - F1(W1)
C2 = F2(C1,W2)
2036
C3 = F3(C2,W3)

Fig. 3G

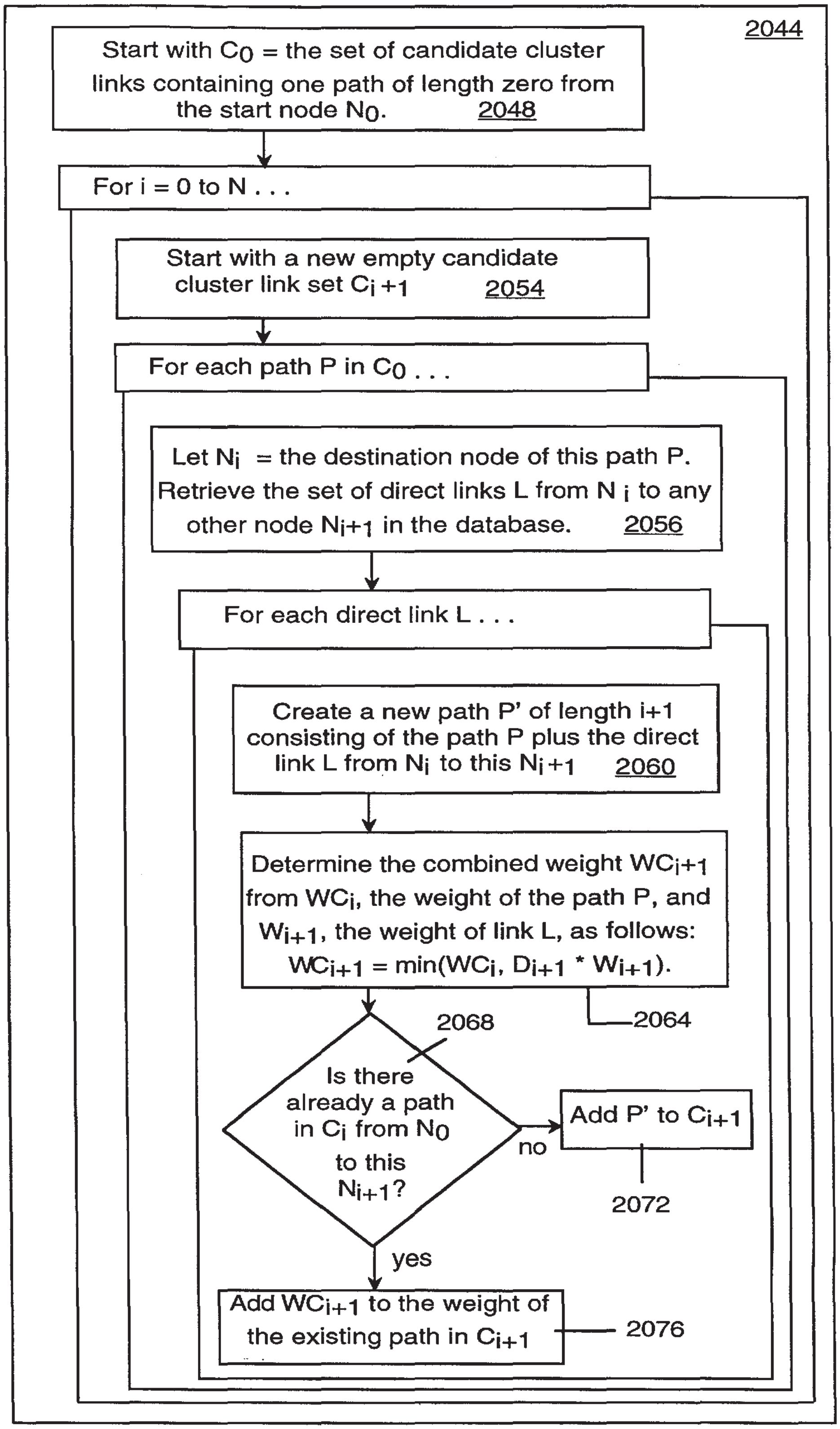

2044
Start with C_0 = the set of candidate cluster links containing one path of length zero from the start node N_0. 2048
For i = 0 to N . . .
Start with a new empty candidate cluster link set C_i+1 2054
For each path P in C_0 . . .
Let N_i = the destination node of this path P. Retrieve the set of direct links L from N_i to any other node N_{i+1} in the database. 2056
For each direct link L . . .
Create a new path P' of length i+1 consisting of the path P plus the direct link L from N_i to this N_i+1 2060
Determine the combined weight WC_{i+1} from WC_i, the weight of the path P, and W_{i+1}, the weight of link L, as follows: $WC_{i+1} = \min(WC_i, D_{i+1} * W_{i+1})$.
2064
2068
Is there already a path in C_i from N_0 to this N_{i+1}?
no
Add P' to C_{i+1}
2072
yes
Add WC_{i+1} to the weight of the existing path in C_{i+1}
2076

Fig. 3H

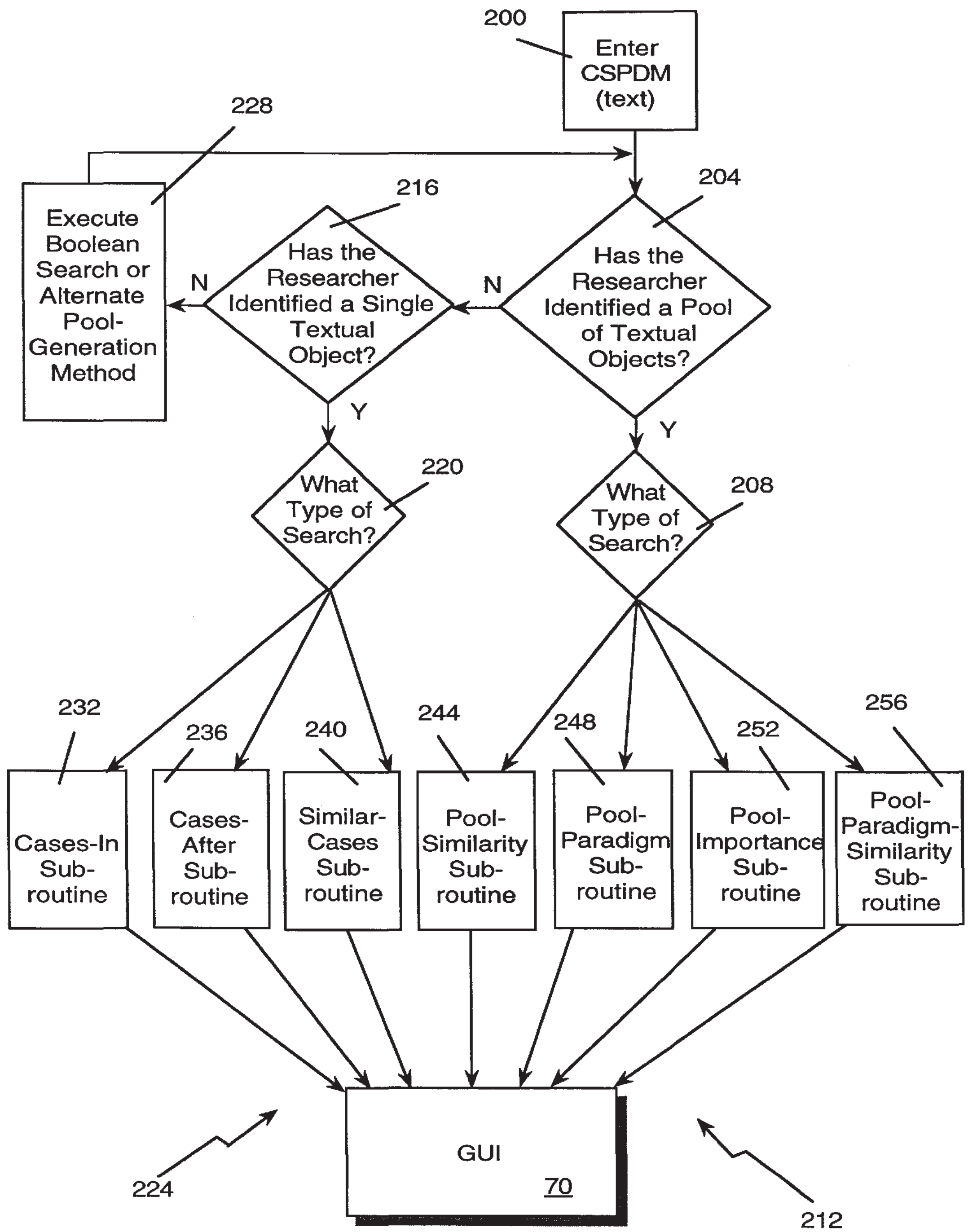
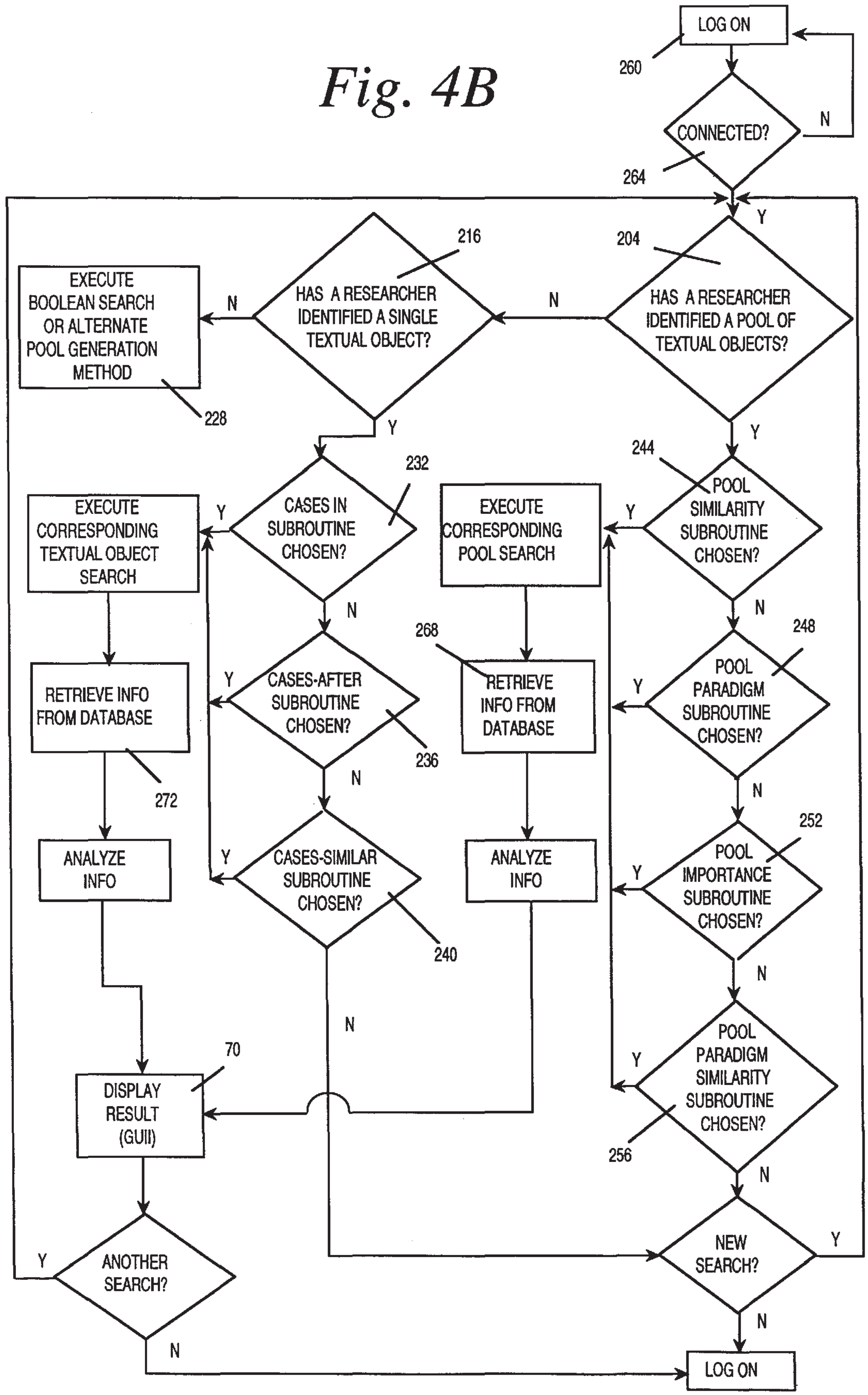

200
Enter CSPDM (text)
228
216
204
Execute Boolean Search or Alternate Pool-Generation Method
N
Has the Researcher Identified a Single Textual Object?
N
Has the Researcher Identified a Pool of Textual Objects?
Y
Y
220
What Type of Search?
What Type of Search?
208
232
236
240
244
248
252
256
Cases-In Sub-routine
Cases-After Sub-routine
Similar-Cases Sub-routine
Pool-Similarity Sub-routine
Pool-Paradigm Sub-routine
Pool-Importance Sub-routine
Pool-Paradigm-Similarity Sub-routine
GUI
70
224
212

Fig. 4A

Fig. 4B



LOG ON
260
CONNECTED?
N
264
Y
204
HAS A RESEARCHER IDENTIFIED A POOL OF TEXTUAL OBJECTS?
N
216
HAS A RESEARCHER IDENTIFIED A SINGLE TEXTUAL OBJECT?
N
EXECUTE BOOLEAN SEARCH OR ALTERNATE POOL GENERATION METHOD
228
Y
232
CASES IN SUBROUTINE CHOSEN?
Y
EXECUTE CORRESPONDING TEXTUAL OBJECT SEARCH
N
CASES-AFTER SUBROUTINE CHOSEN?
236
RETRIEVE INFO FROM DATABASE
272
CASES-SIMILAR SUBROUTINE CHOSEN?
240
ANALYZE INFO
244
POOL SIMILARITY SUBROUTINE CHOSEN?
EXECUTE CORRESPONDING POOL SEARCH
268
RETRIEVE INFO FROM DATABASE
248
POOL PARADIGM SUBROUTINE CHOSEN?
252
POOL IMPORTANCE SUBROUTINE CHOSEN?
ANALYZE INFO
POOL PARADIGM SIMILARITY SUBROUTINE CHOSEN?
256
70
DISPLAY RESULT (GUII)
NEW SEARCH?
ANOTHER SEARCH?
LOG ON

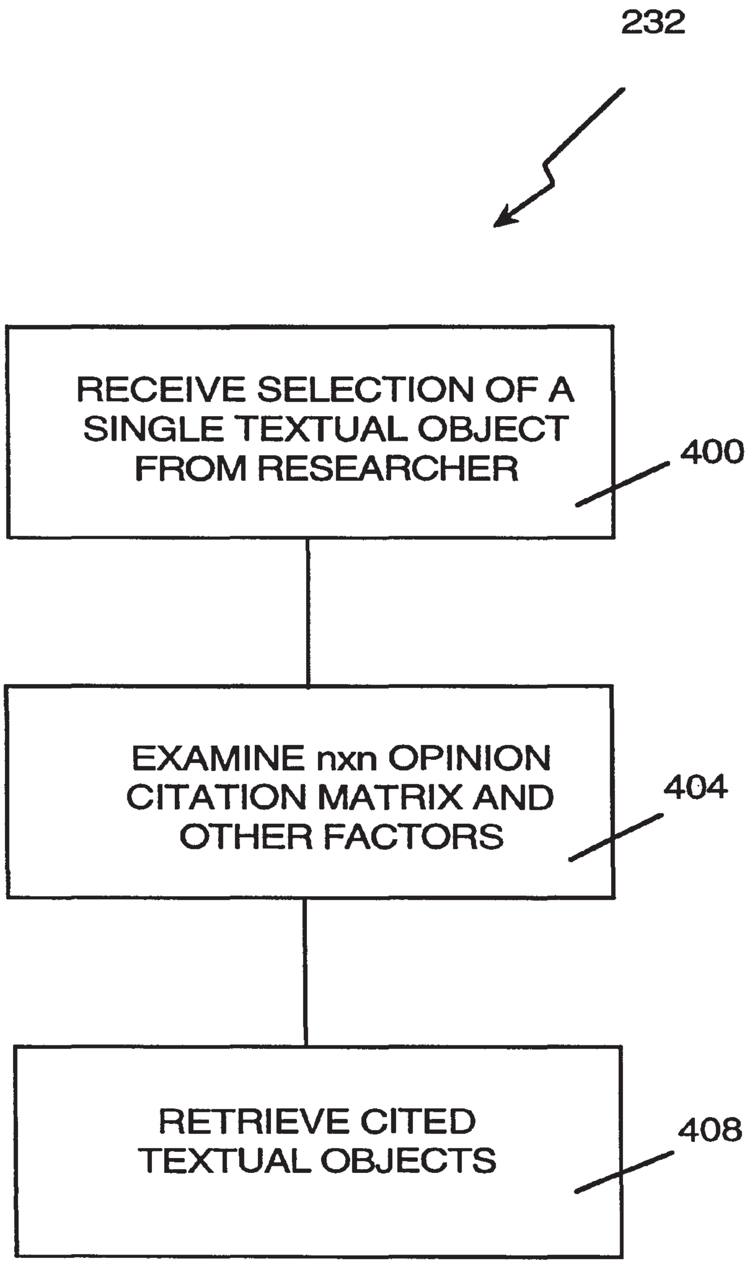

232
RECEIVE SELECTION OF A SINGLE TEXTUAL OBJECT FROM RESEARCHER
400
EXAMINE nxn OPINION CITATION MATRIX AND OTHER FACTORS
404
RETRIEVE CITED TEXTUAL OBJECTS
408

Fig. 4C

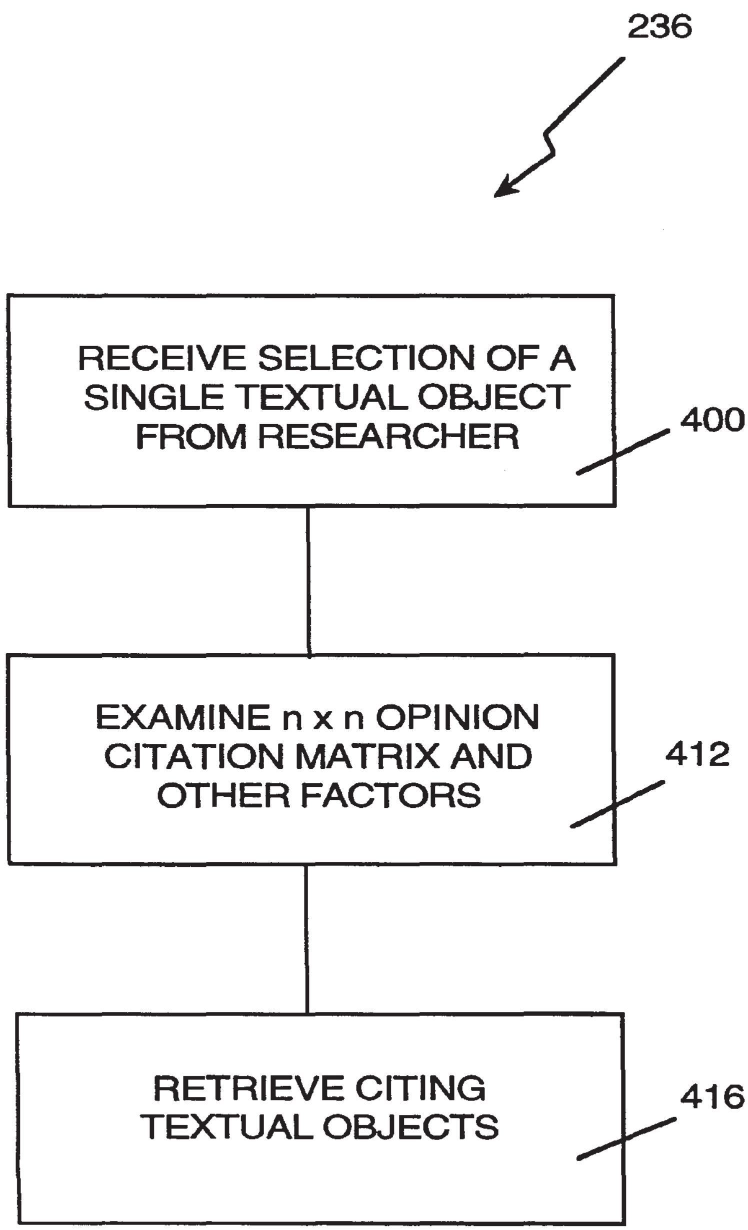

236
RECEIVE SELECTION OF A SINGLE TEXTUAL OBJECT FROM RESEARCHER
400
EXAMINE n x n OPINION CITATION MATRIX AND OTHER FACTORS
412
RETRIEVE CITING TEXTUAL OBJECTS
416

Fig. 4D

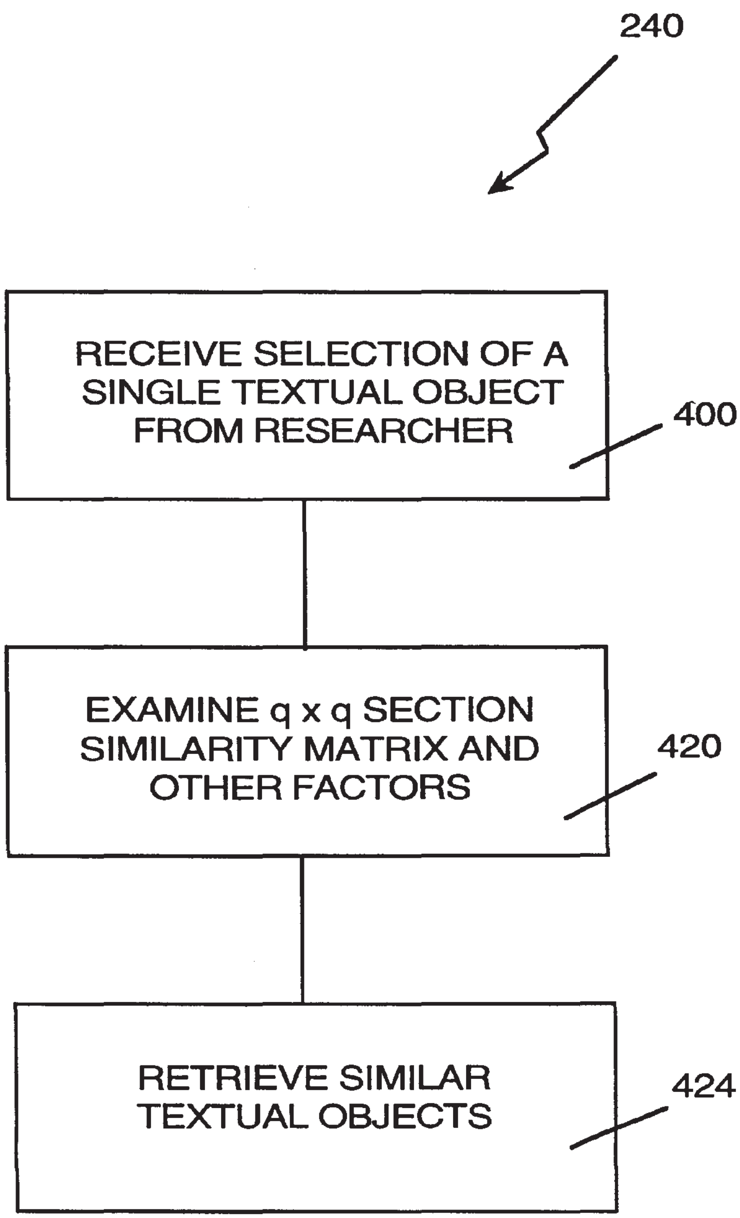
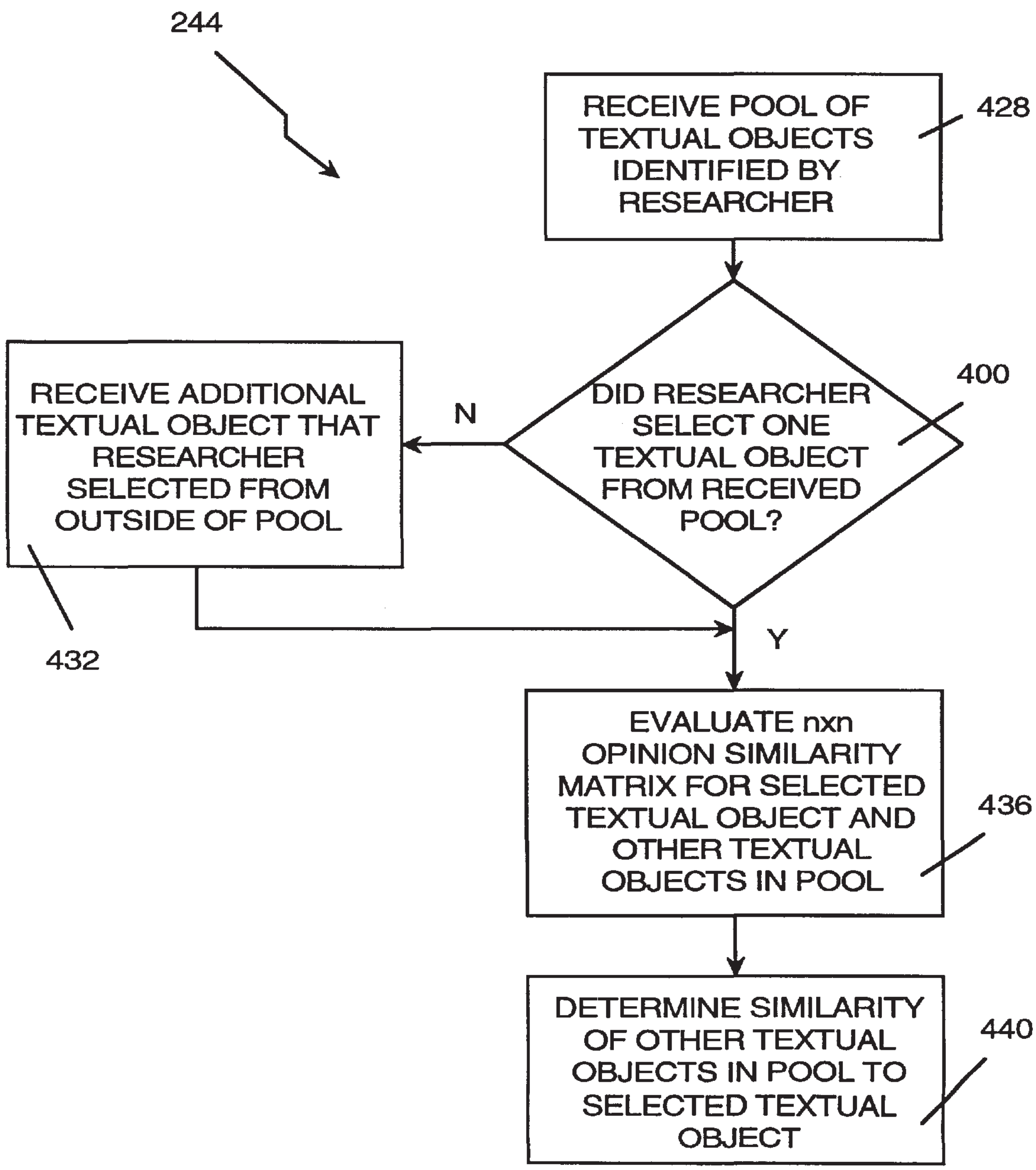

240
RECEIVE SELECTION OF A SINGLE TEXTUAL OBJECT FROM RESEARCHER
400
EXAMINE q x q SECTION SIMILARITY MATRIX AND OTHER FACTORS
420
RETRIEVE SIMILAR TEXTUAL OBJECTS
424

Fig. 4E



244
RECEIVE POOL OF TEXTUAL OBJECTS IDENTIFIED BY RESEARCHER
428
DID RESEARCHER SELECT ONE TEXTUAL OBJECT FROM RECEIVED POOL?
400
N
RECEIVE ADDITIONAL TEXTUAL OBJECT THAT RESEARCHER SELECTED FROM OUTSIDE OF POOL
432
Y
EVALUATE nxn OPINION SIMILARITY MATRIX FOR SELECTED TEXTUAL OBJECT AND OTHER TEXTUAL OBJECTS IN POOL
436
DETERMINE SIMILARITY OF OTHER TEXTUAL OBJECTS IN POOL TO SELECTED TEXTUAL OBJECT
440

Fig. 4F

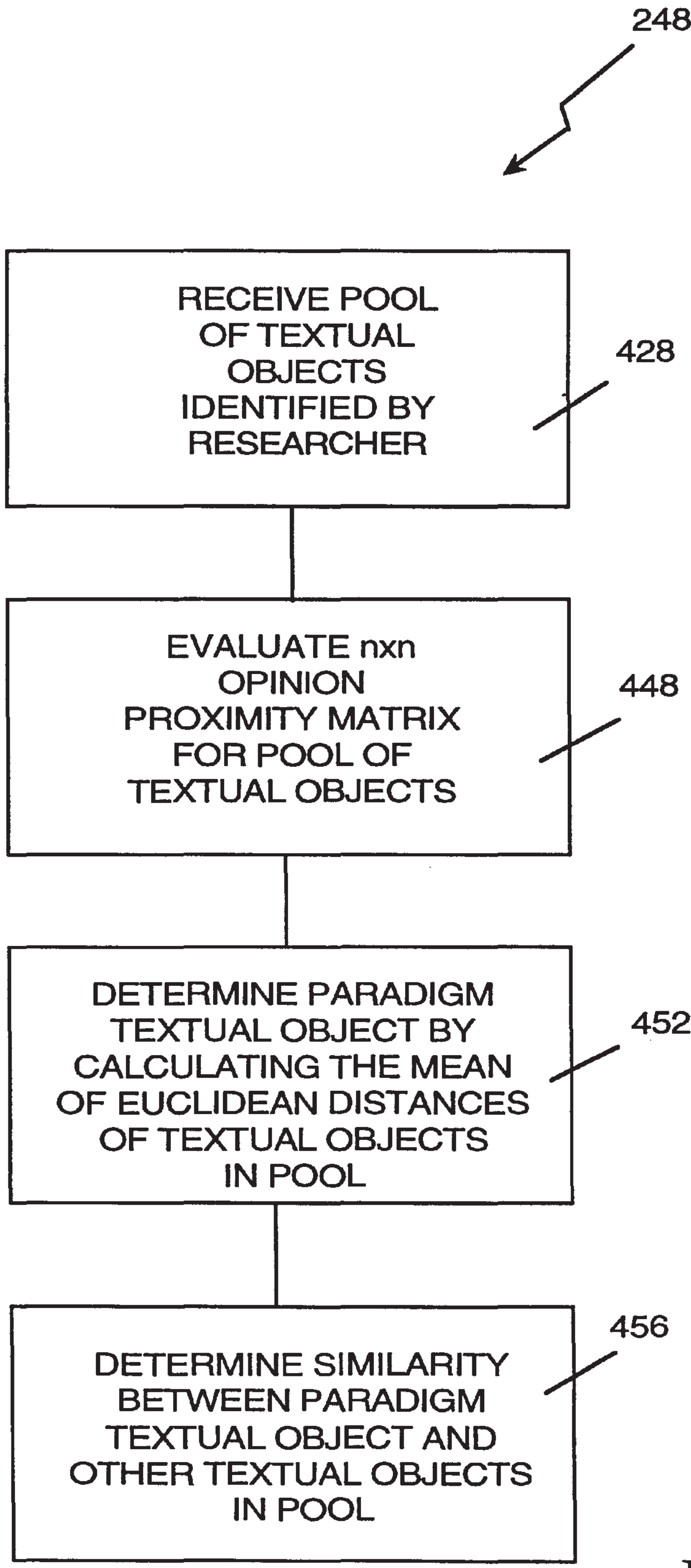
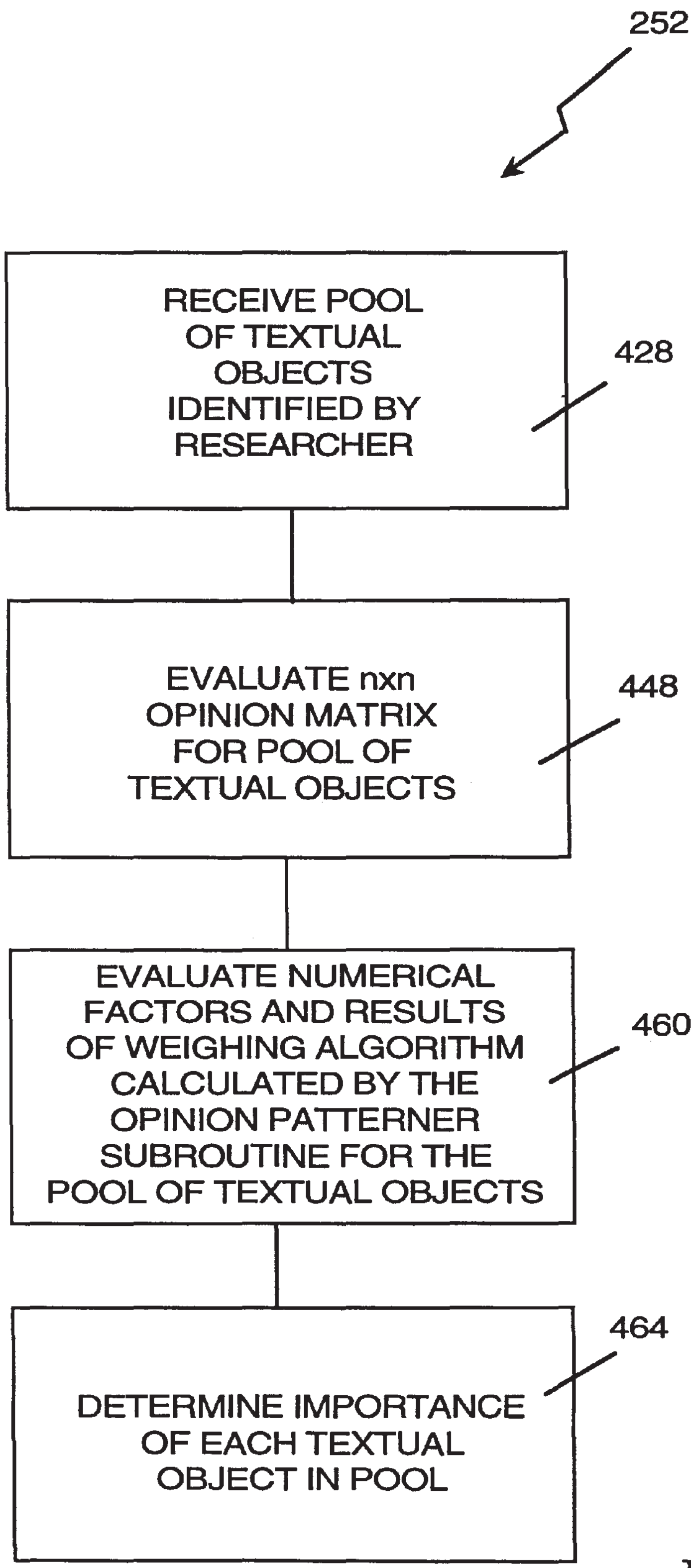

248
RECEIVE POOL OF TEXTUAL OBJECTS IDENTIFIED BY RESEARCHER
428
EVALUATE nxn OPINION PROXIMITY MATRIX FOR POOL OF TEXTUAL OBJECTS
448
DETERMINE PARADIGM TEXTUAL OBJECT BY CALCULATING THE MEAN OF EUCLIDEAN DISTANCES OF TEXTUAL OBJECTS IN POOL
452
DETERMINE SIMILARITY BETWEEN PARADIGM TEXTUAL OBJECT AND OTHER TEXTUAL OBJECTS IN POOL
456

Fig. 4G



252
RECEIVE POOL OF TEXTUAL OBJECTS IDENTIFIED BY RESEARCHER
428
EVALUATE nxn OPINION MATRIX FOR POOL OF TEXTUAL OBJECTS
448
EVALUATE NUMERICAL FACTORS AND RESULTS OF WEIGHING ALGORITHM CALCULATED BY THE OPINION PATTERNER SUBROUTINE FOR THE POOL OF TEXTUAL OBJECTS
460
DETERMINE IMPORTANCE OF EACH TEXTUAL OBJECT IN POOL
464

Fig. 4H

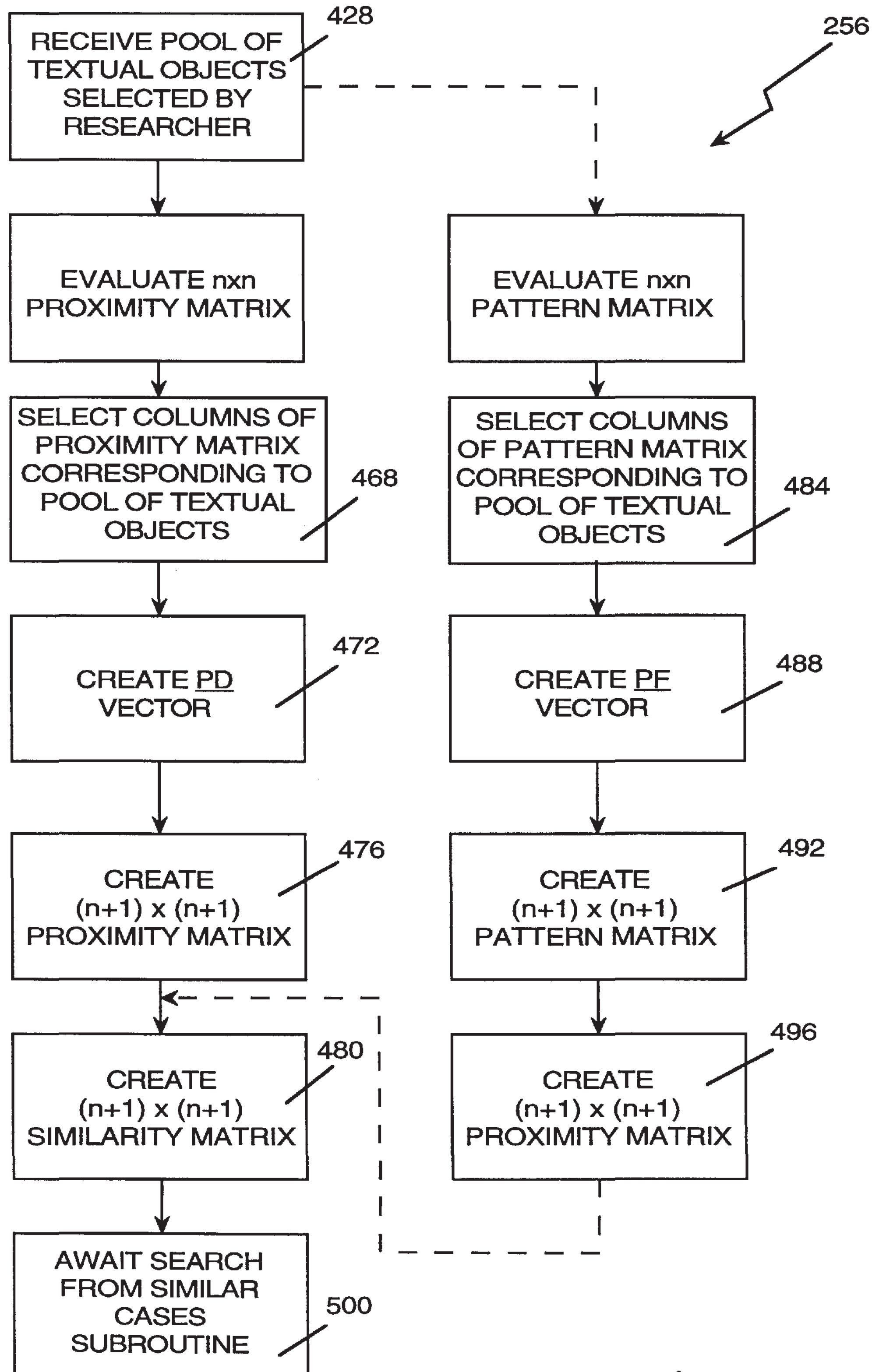

256
RECEIVE POOL OF TEXTUAL OBJECTS SELECTED BY RESEARCHER
428
EVALUATE nxn PROXIMITY MATRIX
EVALUATE nxn PATTERN MATRIX
SELECT COLUMNS OF PROXIMITY MATRIX CORRESPONDING TO POOL OF TEXTUAL OBJECTS
468
SELECT COLUMNS OF PATTERN MATRIX CORRESPONDING TO POOL OF TEXTUAL OBJECTS
484
CREATE PD VECTOR
472
CREATE PF VECTOR
488
CREATE (n+1) x (n+1) PROXIMITY MATRIX
476
CREATE (n+1) x (n+1) PATTERN MATRIX
492
CREATE (n+1) x (n+1) SIMILARITY MATRIX
480
CREATE (n+1) x (n+1) PROXIMITY MATRIX
496
AWAIT SEARCH FROM SIMILAR CASES SUBROUTINE
500

Fig. 4I

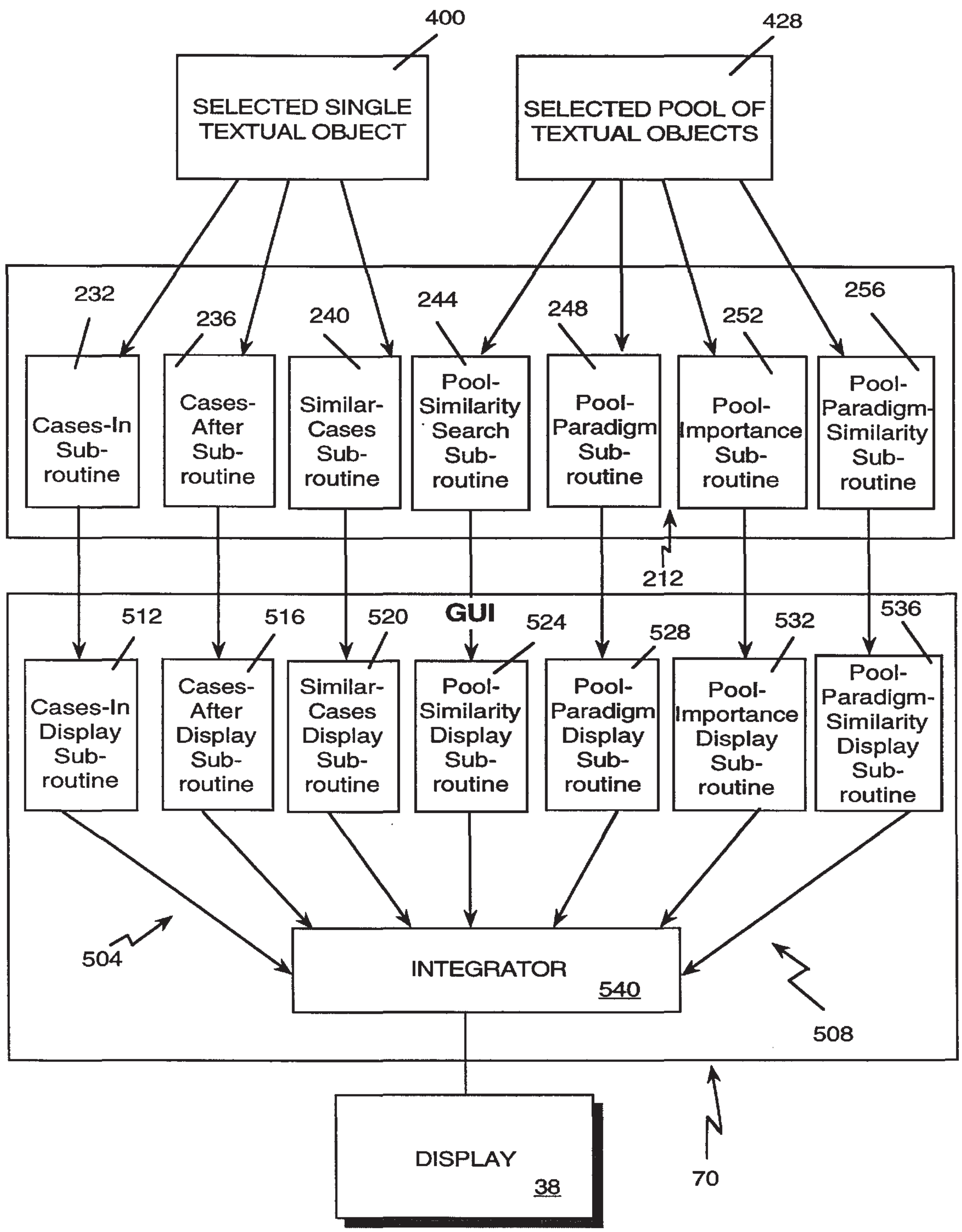

400
428
SELECTED SINGLE TEXTUAL OBJECT
SELECTED POOL OF TEXTUAL OBJECTS
232
236
240
244
248
252
256
Cases-In Sub-routine
Cases-After Sub-routine
Similar-Cases Sub-routine
Pool-Similarity Search Sub-routine
Pool-Paradigm Sub-routine
Pool-Importance Sub-routine
Pool-Paradigm-Similarity Sub-routine
212
GUI
512
516
520
524
528
532
536
Cases-In Display Sub-routine
Cases-After Display Sub-routine
Similar-Cases Display Sub-routine
Pool-Similarity Display Sub-routine
Pool-Paradigm Display Sub-routine
Pool-Importance Display Sub-routine
Pool-Paradigm-Similarity Display Sub-routine
504
INTEGRATOR 540
508
DISPLAY 38
70

Fig. 5A

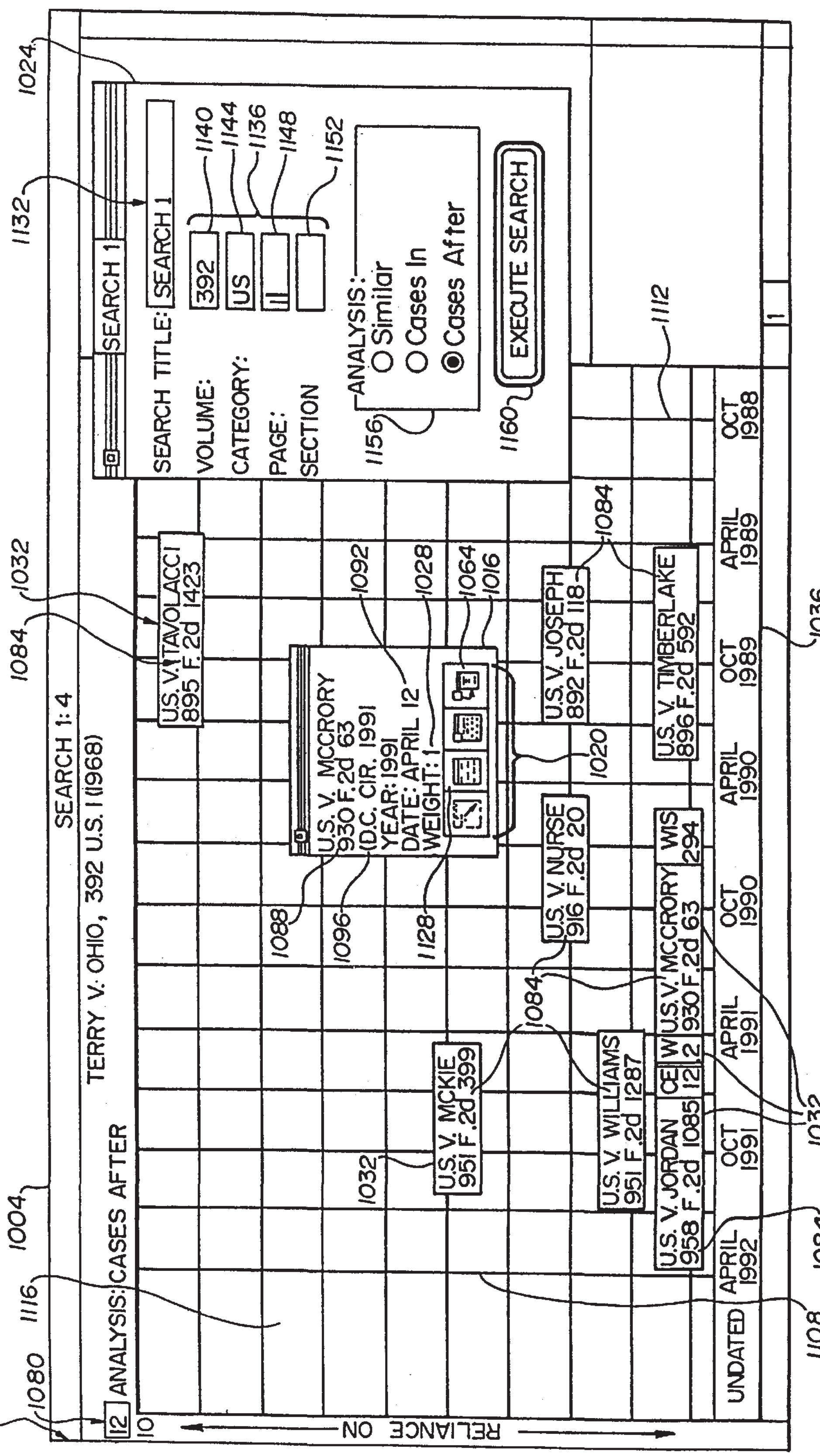
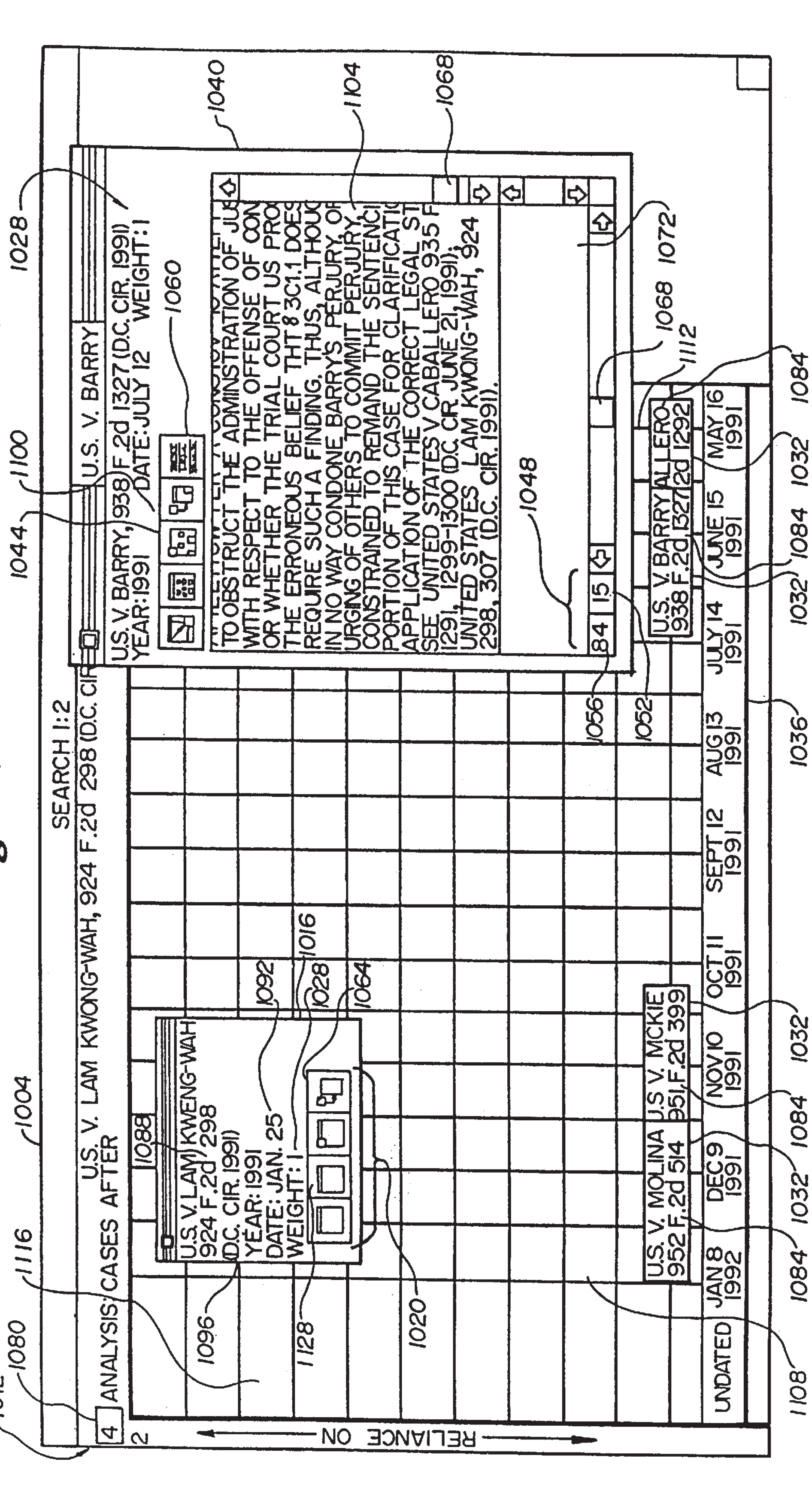


Fig. 5B

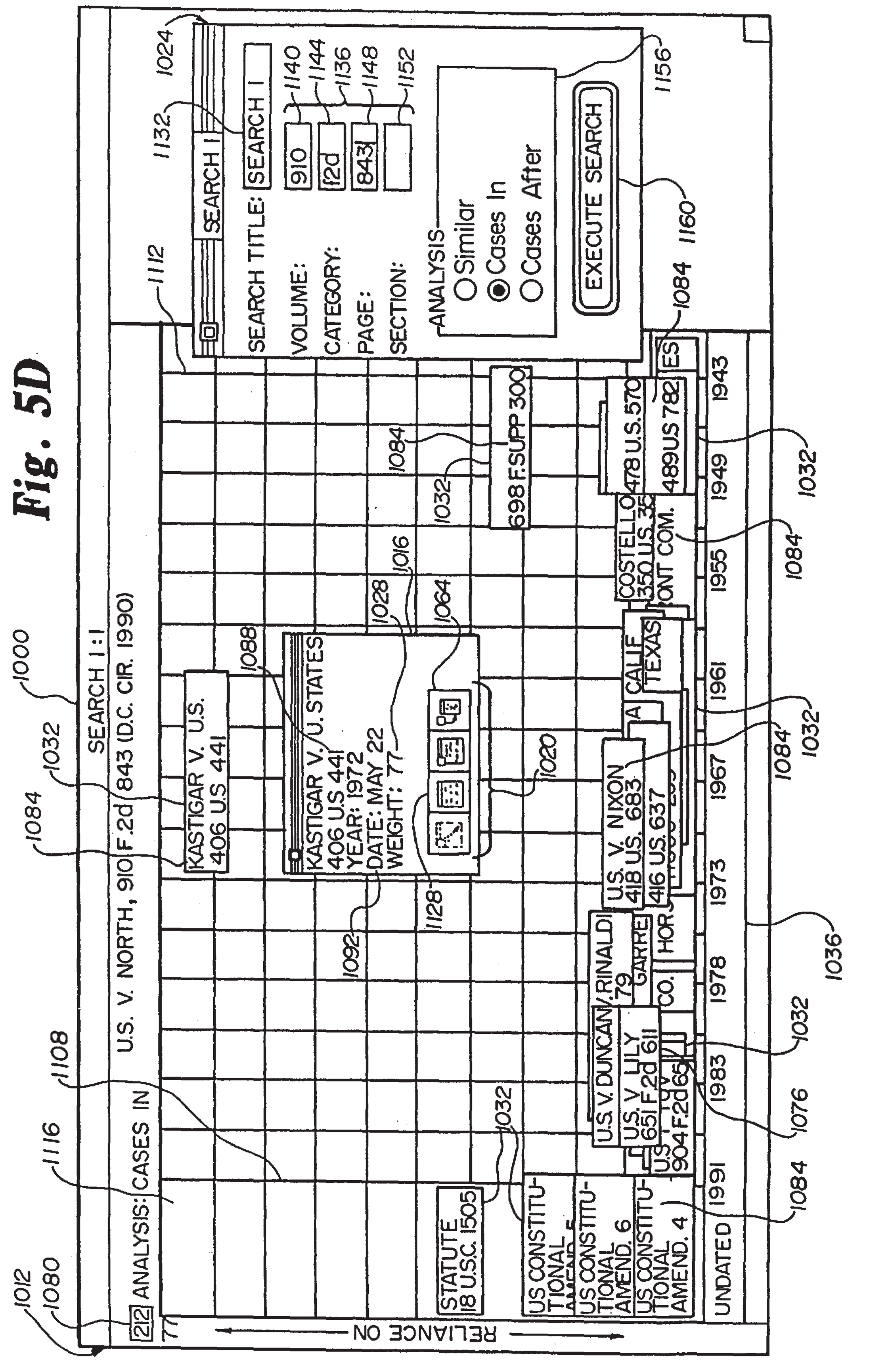
Facebook Inc. Ex. 1001

022

Fig. 5C



ANALYSIS: CASES AFTER
U.S. V. LAM KWONG-WAH, 924 F.2d 298 (D.C. CIR
SEARCH 1:2
U.S. V. BARRY
U.S. V. BARRY, 938 F.2d 1327 (D.C. CIR. 1991)
YEAR: 1991 DATE: JULY 12 WEIGHT: 1
U.S. V. LAM KWENG-WAH
924 F.2d 298
D.C. CIR. 1991
YEAR: 1991
DATE: JAN. 25
WEIGHT: 1
U.S. V. MOLINA 952 F.2d 514
U.S V. MCKIE 951 F.2d 399
U.S. V. BARRY 938 F.2d 1327
CABALLERO 2d 1292
UNDATED
JAN 8 1992
DEC 9 1991
NOV 10 1991
OCT 11 1991
SEPT 12 1991
AUG 13 1991
JULY 14 1991
JUNE 15 1991
MAY 16 1991
RELIANCE ON



U.S. V. NORTH, 910 F.2d 843 (D.C. CIR. 1990)
SEARCH 1:1
ANALYSIS: CASES IN
RELIANCE ON
SEARCH 1
SEARCH TITLE: SEARCH 1
VOLUME: 910
CATEGORY: F2d
PAGE: 843
SECTION:
ANALYSIS
Similar
Cases In
Cases After
EXECUTE SEARCH
KASTIGAR V. U.S. 406 U.S 441
KASTIGAR V. U. STATES
406 U.S 441
YEAR: 1972
DATE: MAY 22
WEIGHT: 77
698 F.SUPP 300
478 U.S.570
489 US 782
U.S. V. NIXON 418 U.S. 683
416 U.S. 637
U.S V. DUNCAN
U.S. V. LILY 651 F.2d 611
904 F.2d 65
STATUTE 18 U.S.C. 1505
US CONSTITUTIONAL AMEND. 6
US CONSTITUTIONAL AMEND. 4
UNDATED
1991
1983
1978
1973
1967
1961
1955
1949
1943

Fig. 5D

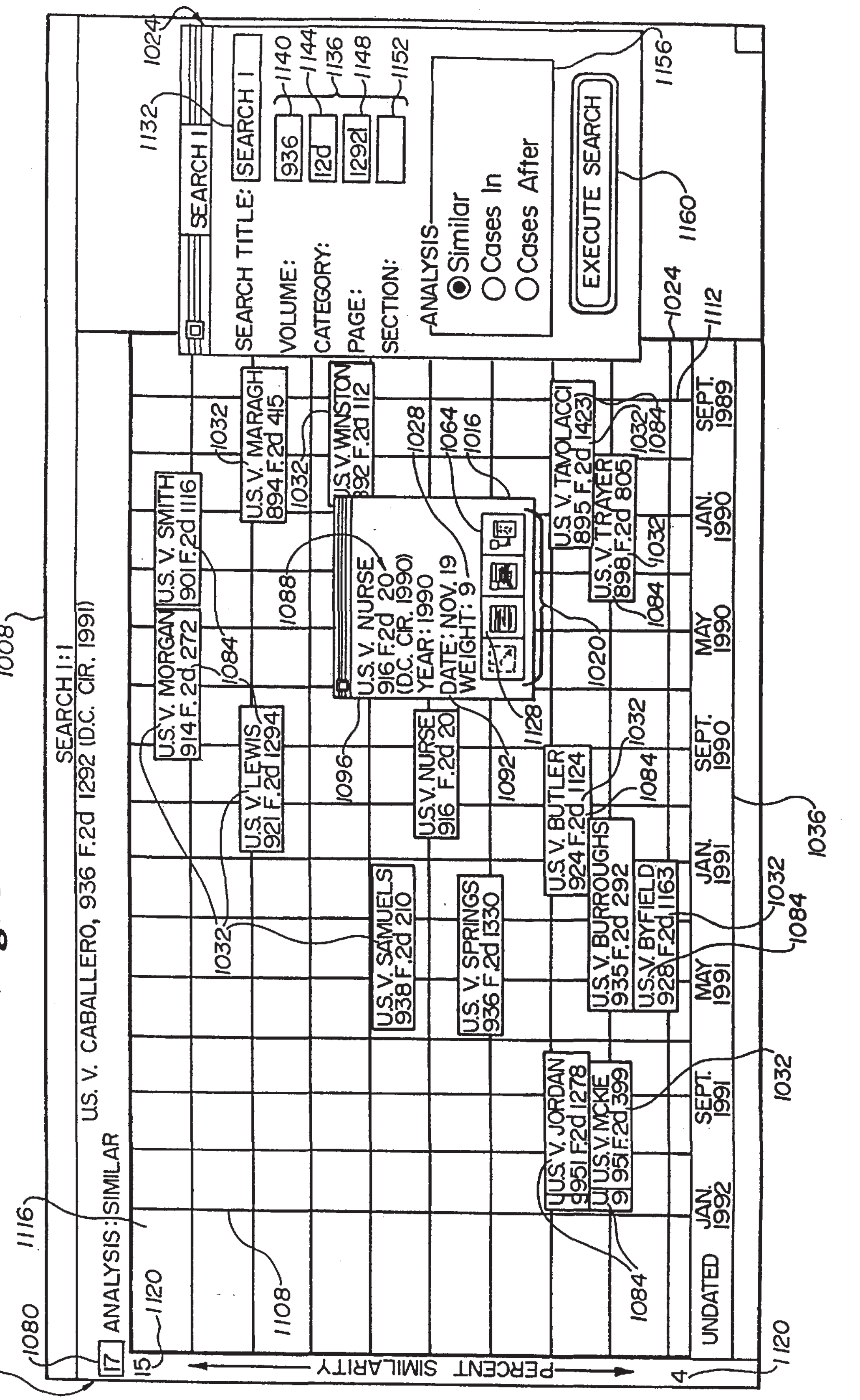
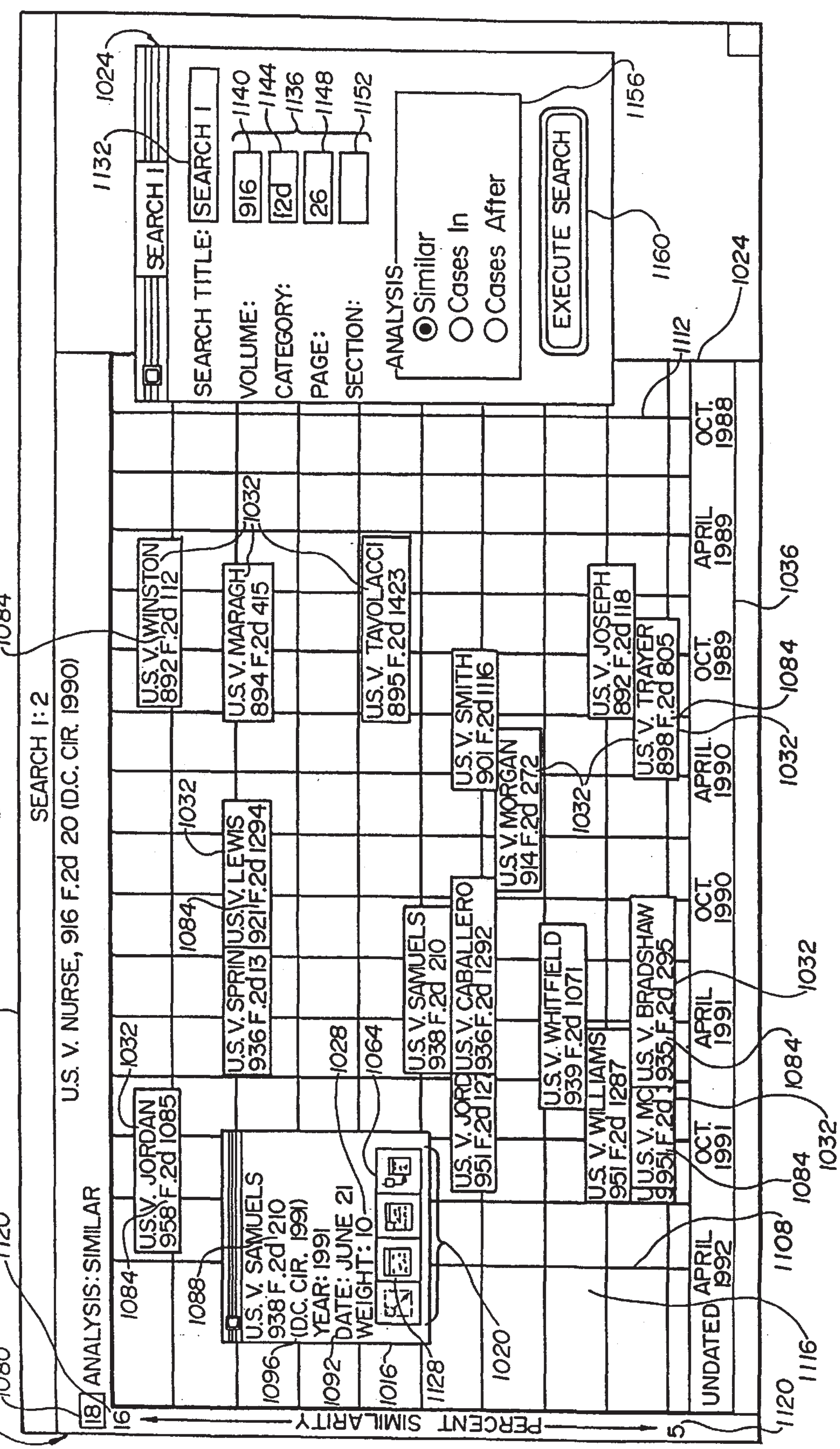

SEARCH 1:1
U.S. V. CABALLERO, 936 F.2d 1292 (D.C. CIR. 1991)
ANALYSIS: SIMILAR
SEARCH TITLE: SEARCH 1
VOLUME: 936
CATEGORY: 12d
PAGE: 1292
SECTION:
ANALYSIS
Similar
Cases In
Cases After
EXECUTE SEARCH
U.S. V. NURSE
916 F.2d 20
(D.C. CIR. 1990)
YEAR: 1990
DATE: NOV. 19
WEIGHT: 9
PERCENT SIMILARITY
UNDATED
JAN. 1992
SEPT. 1991
MAY 1991
JAN. 1991
SEPT. 1990
MAY 1990
JAN. 1990
SEPT. 1989

Fig. 5E

Fig. 5F



U.S. V. NURSE, 916 F.2d 20 (D.C. CIR. 1990)
SEARCH 1:2
ANALYSIS: SIMILAR
SEARCH 1
SEARCH TITLE: SEARCH 1
VOLUME: 916
CATEGORY: 12d
PAGE: 26
SECTION:
ANALYSIS
Similar
Cases In
Cases After
EXECUTE SEARCH
U.S.V. JORDAN 958 F.2d 1085
U.S.V. SAMUELS 938 F.2d 210 (D.C. CIR. 1991) YEAR: 1991 DATE: JUNE 21 WEIGHT: 10
U.S.V. WINSTON 892 F.2d 112
U.S.V. MARAGH 894 F.2d 415
U.S.V. TAYOLACCI 895 F.2d 1423
U.S.V. SMITH 901 F.2d 1116
U.S.V. JOSEPH 892 F.2d 118
U.S.V. TRAYER 898 F.2d 805
U.S.V. MORGAN 914 F.2d 272
U.S.V. LEWIS 921 F.2d 1294
U.S.V. SPRIN 936 F.2d 13
U.S.V. SAMUELS 938 F.2d 210
U.S.V. CABALLERO 936 F.2d 1292
U.S.V. JORD 951 F.2d 12
U.S.V. WHITFIELD 939 F.2d 1071
U.S.V. WILLIAMS 951 F.2d 1287
U.S.V. MC 995 F.2d
U.S.V. BRADSHAW 935 F.2d 295
UNDATED
APRIL 1992
OCT. 1991
APRIL 1991
OCT. 1990
APRIL 1990
OCT. 1989
APRIL 1989
OCT. 1988
PERCENT SIMILARITY
18
16
5
1008 1012 1016 1020 1024 1028 1032 1036 1064 1080 1084 1088 1092 1096 1108 1112 1116 1120 1128 1132 1136 1140 1144 1148 1152 1156 1160

Fig. 5G

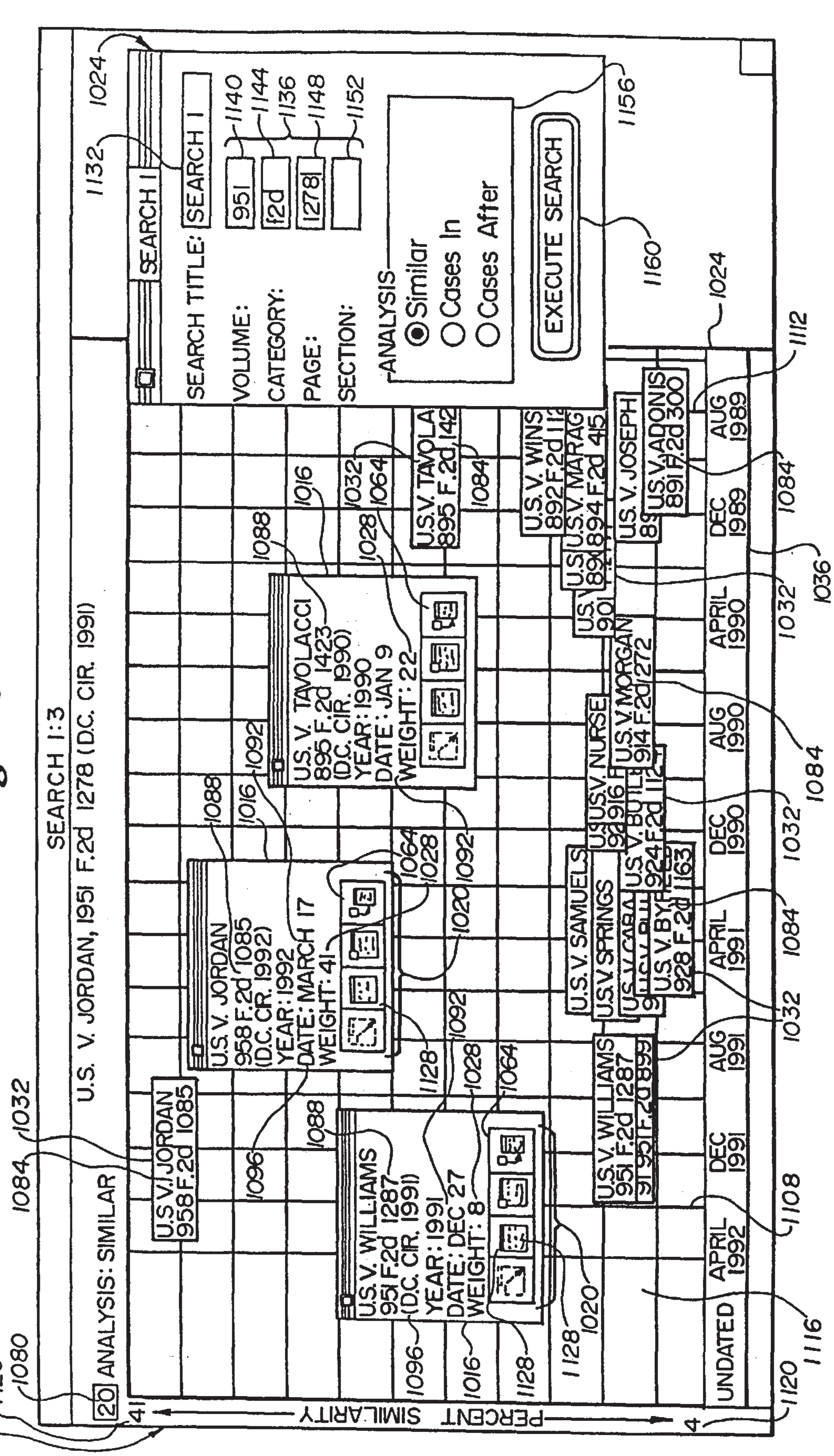
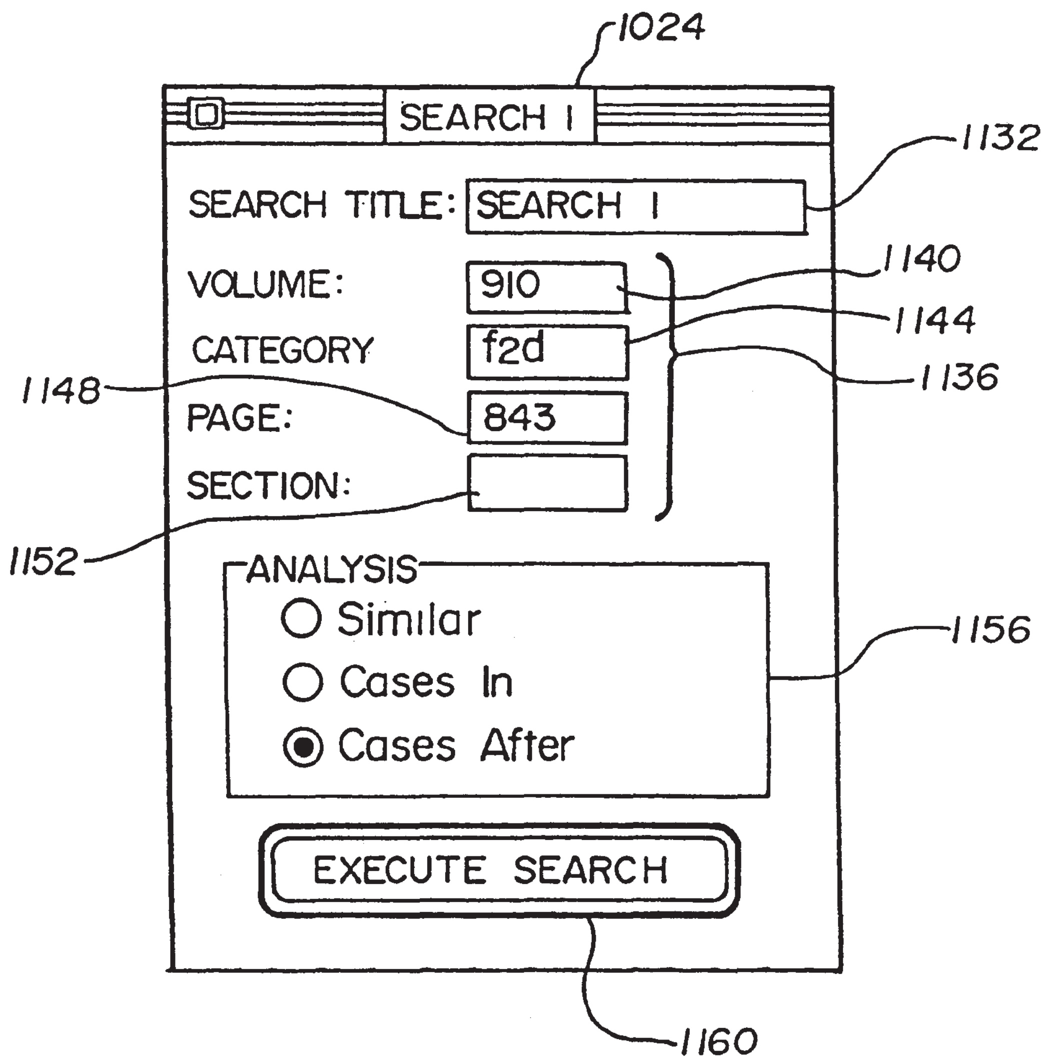

SEARCH 1:3
U.S. V. JORDAN, 1951 F.2d 1278 (D.C. CIR. 1991)
20 ANALYSIS: SIMILAR
SEARCH 1
SEARCH TITLE: SEARCH 1
VOLUME: 951
CATEGORY: f2d
PAGE: 1278
SECTION:
ANALYSIS
Similar
Cases In
Cases After
EXECUTE SEARCH
U.S. V. JORDAN
958 F.2d 1085
(D.C. CIR. 1992)
YEAR: 1992
DATE: MARCH 17
WEIGHT: 41
U.S. V. TAVOLACCI
895 F.2d 1423
(D.C. CIR. 1990)
YEAR: 1990
DATE: JAN 9
WEIGHT: 22
U.S. V. WILLIAMS
951 F.2d 1287
(D.C. CIR. 1991)
YEAR: 1991
DATE: DEC 27
WEIGHT: 8
PERCENT SIMILARITY
41
4
UNDATED
APRIL 1992
DEC 1991
AUG 1991
APRIL 1991
DEC 1990
AUG 1990
APRIL 1990
DEC 1989
AUG 1989

Fig. 5H


1024
SEARCH I
1132
SEARCH TITLE: SEARCH I
1140
VOLUME: 910
1144
CATEGORY f2d
1136
1148
PAGE: 843
SECTION:
1152
ANALYSIS
Similar
Cases In
Cases After
1156
EXECUTE SEARCH
1160

SCHEMATIC REPRESENTATIONS OF THE EIGHTEEN PRIMARY PATTERNS

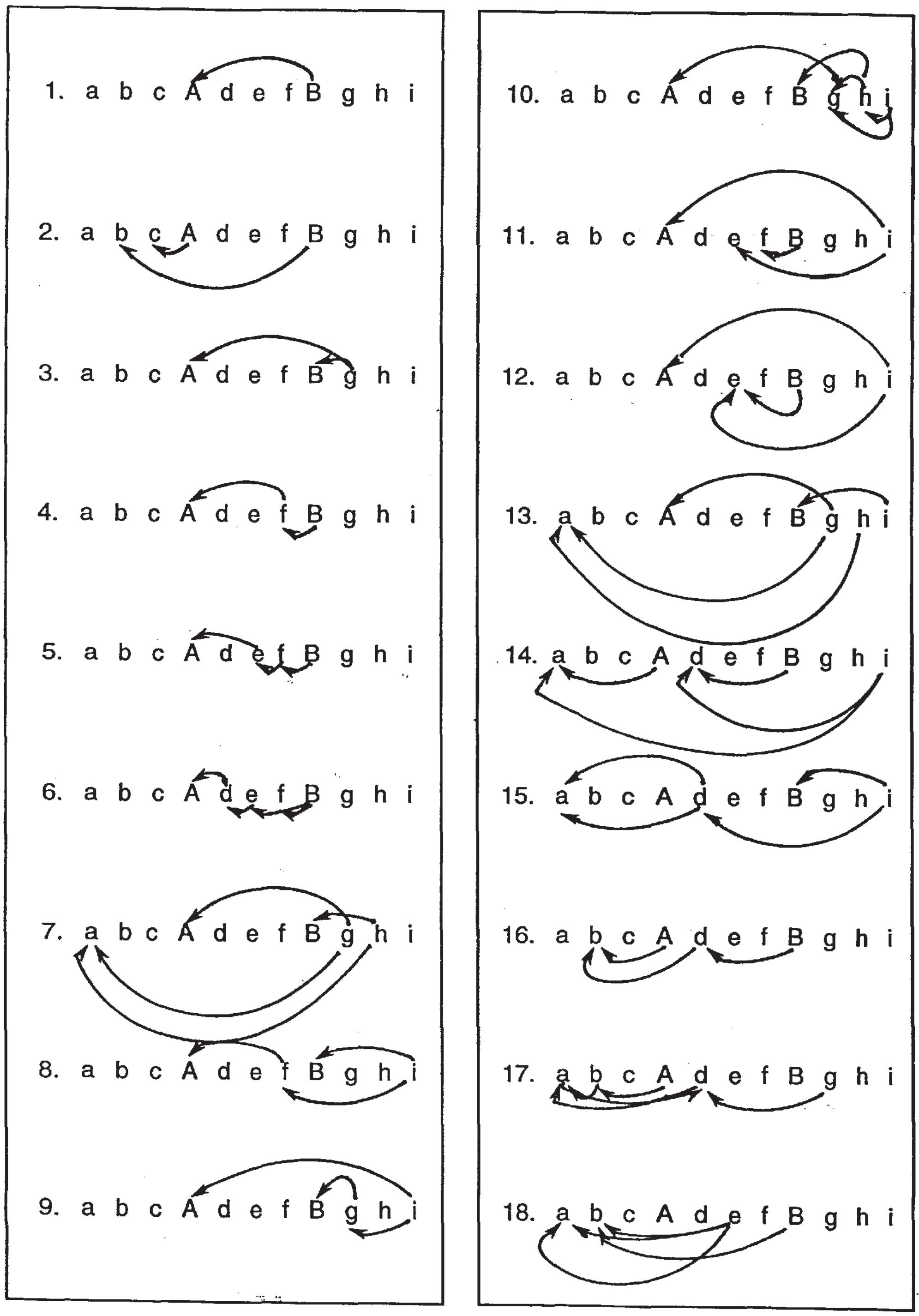


Fig. 6

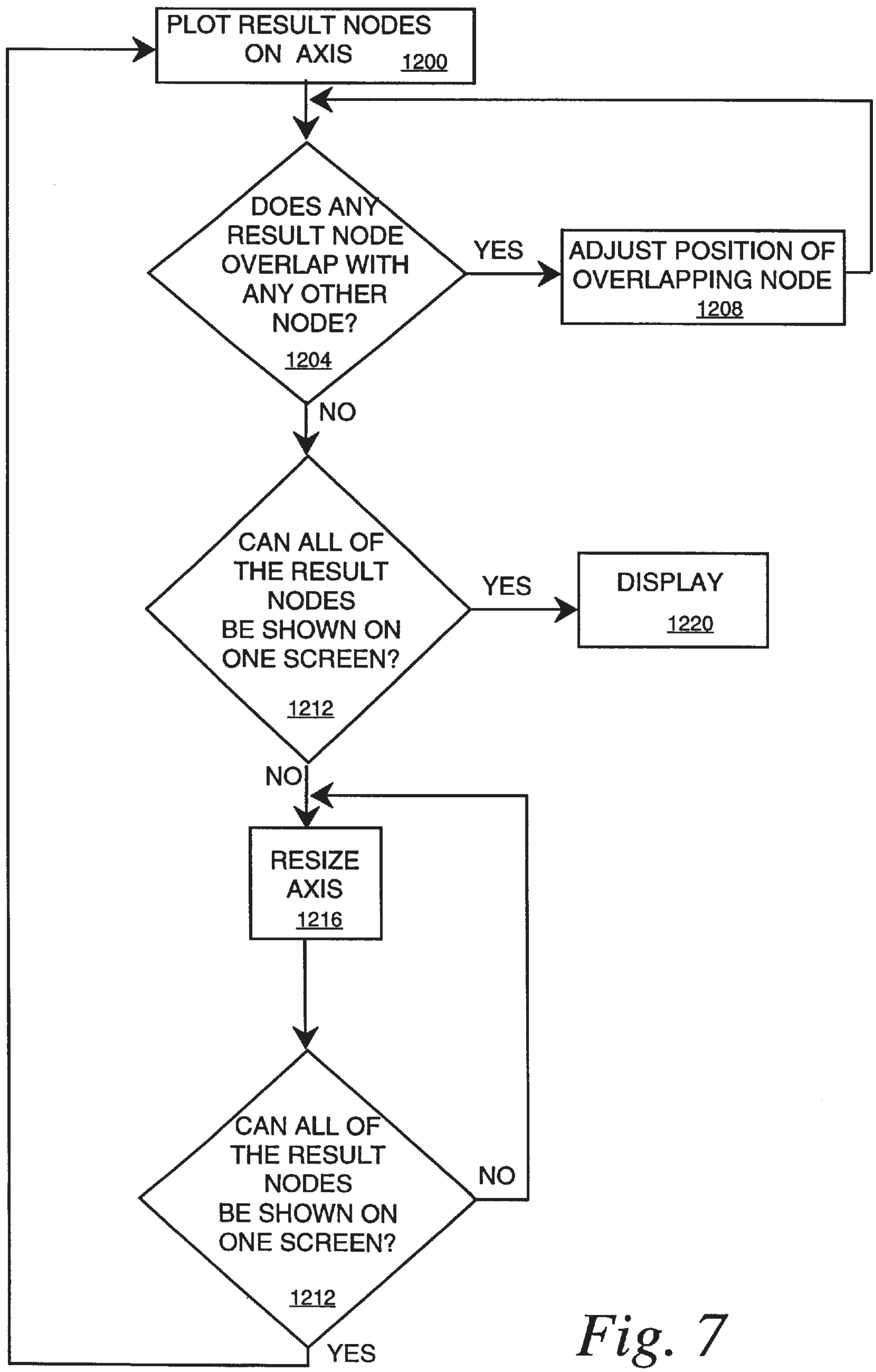

PLOT RESULT NODES ON AXIS 1200
DOES ANY RESULT NODE OVERLAP WITH ANY OTHER NODE? 1204
YES
ADJUST POSITION OF OVERLAPPING NODE 1208
NO
CAN ALL OF THE RESULT NODES BE SHOWN ON ONE SCREEN? 1212
YES
DISPLAY 1220
NO
RESIZE AXIS 1216
CAN ALL OF THE RESULT NODES BE SHOWN ON ONE SCREEN? 1212
NO
YES

Fig. 7

Fig. 8-1

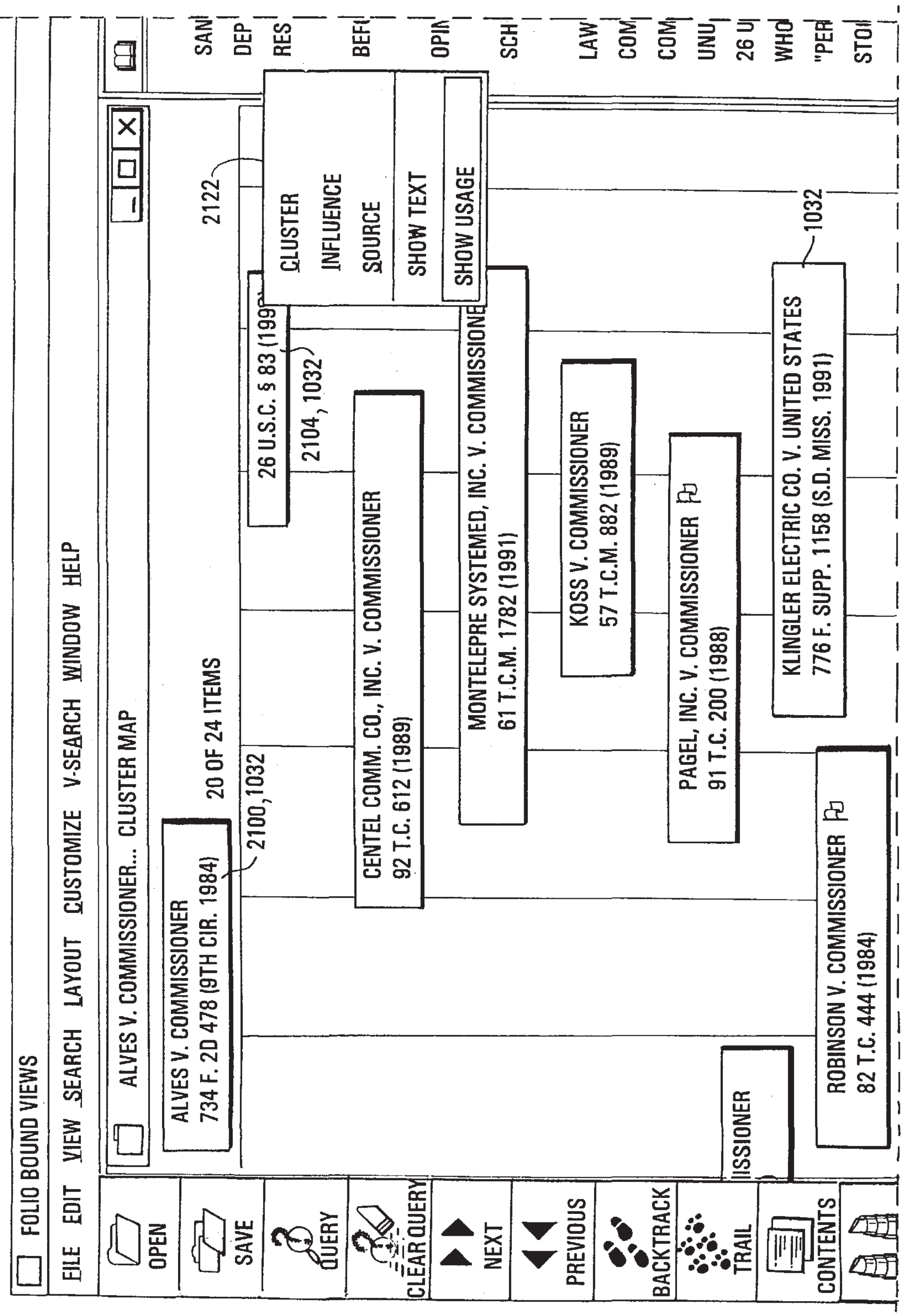
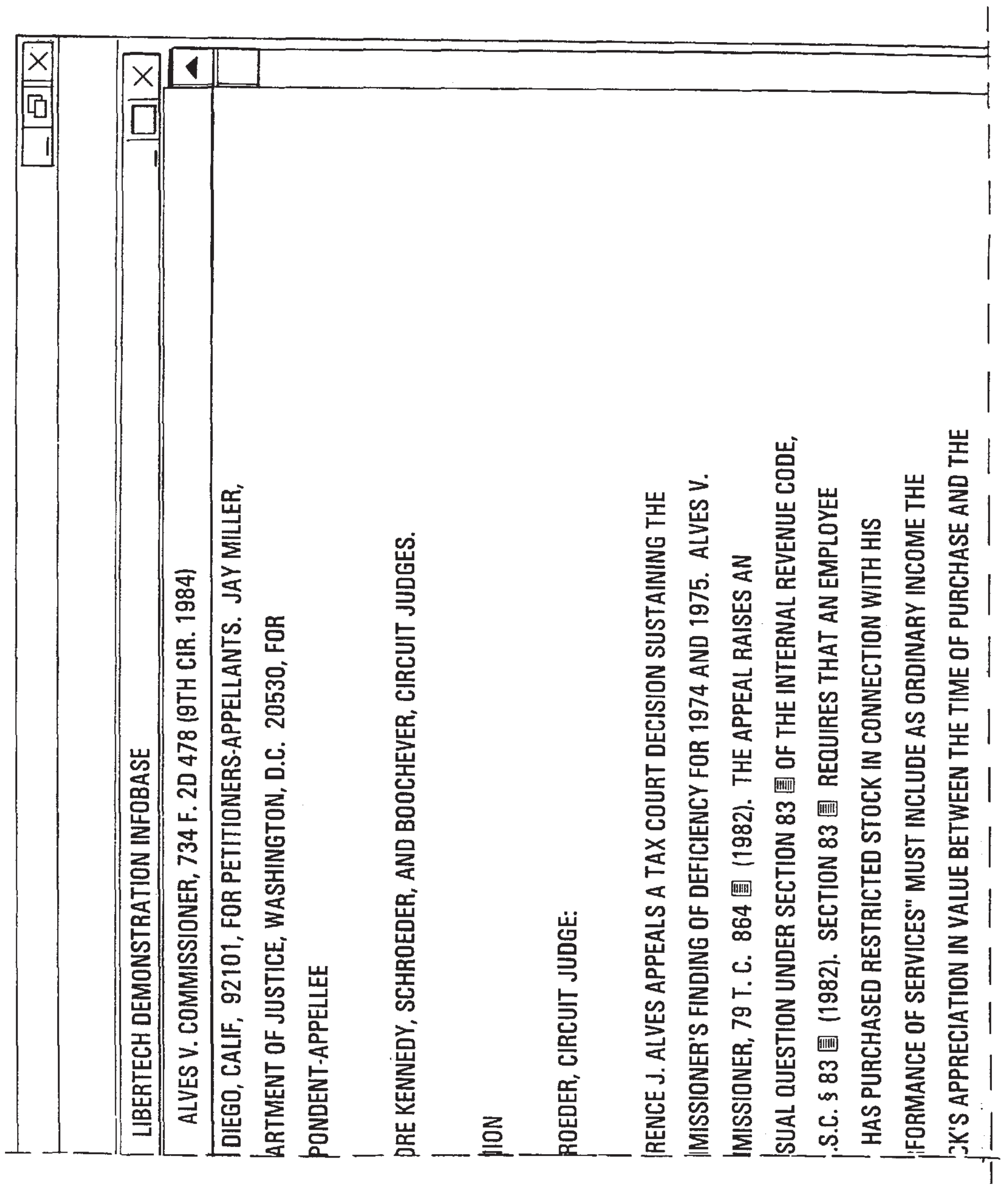

FOLIO BOUND VIEWS
FILE EDIT VIEW SEARCH LAYOUT CUSTOMIZE V-SEARCH WINDOW HELP
ALVES V. COMMISSIONER... CLUSTER MAP
20 OF 24 ITEMS
OPEN
SAVE
QUERY
CLEAR QUERY
NEXT
PREVIOUS
BACKTRACK
TRAIL
CONTENTS
ALVES V. COMMISSIONER
734 F. 2D 478 (9TH CIR. 1984)
2100, 1032
26 U.S.C. § 83 (199
2104, 1032
2122
CLUSTER
INFLUENCE
SOURCE
SHOW TEXT
SHOW USAGE
CENTEL COMM. CO., INC. V. COMMISSIONER
92 T.C. 612 (1989)
MONTELEPRE SYSTEMED, INC. V. COMMISSIONE
61 T.C.M. 1782 (1991)
KOSS V. COMMISSIONER
57 T.C.M. 882 (1989)
PAGEL, INC. V. COMMISSIONER
91 T.C. 200 (1988)
KLINGLER ELECTRIC CO. V. UNITED STATES
776 F. SUPP. 1158 (S.D. MISS. 1991)
1032
ROBINSON V. COMMISSIONER
82 T.C. 444 (1984)
ISSIONER
SAN
DEP
RES
BEF
OPI
SCH
LAW
COM
COM
UNU
26 U
WHO
"PER
STO

Fig. 8-2


LIBERTECH DEMONSTRATION INFOBASE
ALVES V. COMMISSIONER, 734 F. 2D 478 (9TH CIR. 1984)
DIEGO, CALIF, 92101, FOR PETITIONERS-APPELLANTS. JAY MILLER,
ARTMENT OF JUSTICE, WASHINGTON, D.C. 20530, FOR
PONDENT-APPELLEE
ORE KENNEDY, SCHROEDER, AND BOOCHEVER, CIRCUIT JUDGES.
ION
ROEDER, CIRCUIT JUDGE:
RENCE J. ALVES APPEALS A TAX COURT DECISION SUSTAINING THE
MISSIONER'S FINDING OF DEFICIENCY FOR 1974 AND 1975. ALVES V.
MISSIONER, 79 T. C. 864 (1982). THE APPEAL RAISES AN
SUAL QUESTION UNDER SECTION 83 OF THE INTERNAL REVENUE CODE,
.S.C. $ 83 (1982). SECTION 83 REQUIRES THAT AN EMPLOYEE
HAS PURCHASED RESTRICTED STOCK IN CONNECTION WITH HIS
FORMANCE OF SERVICES" MUST INCLUDE AS ORDINARY INCOME THE
K'S APPRECIATION IN VALUE BETWEEN THE TIME OF PURCHASE AND THE

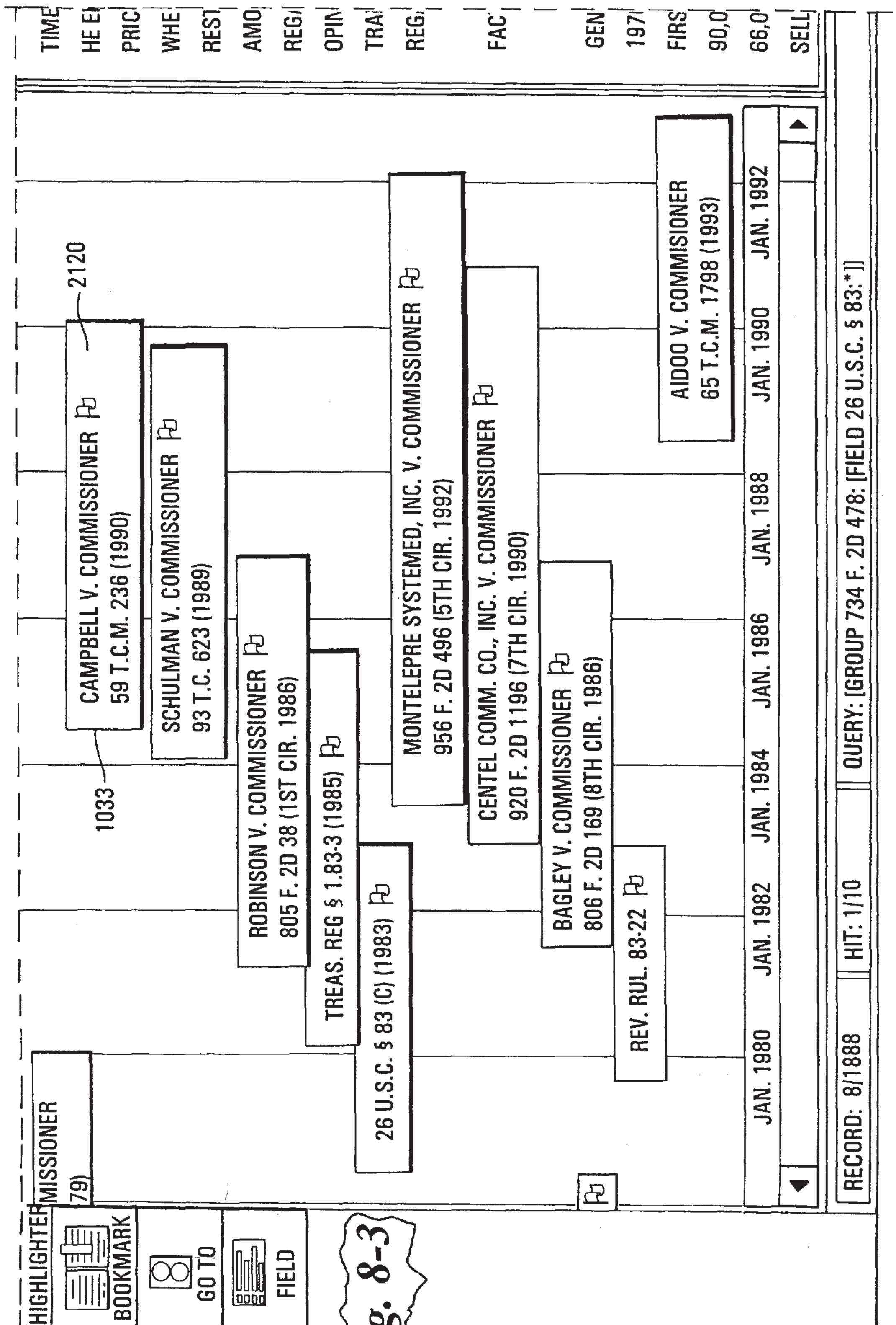
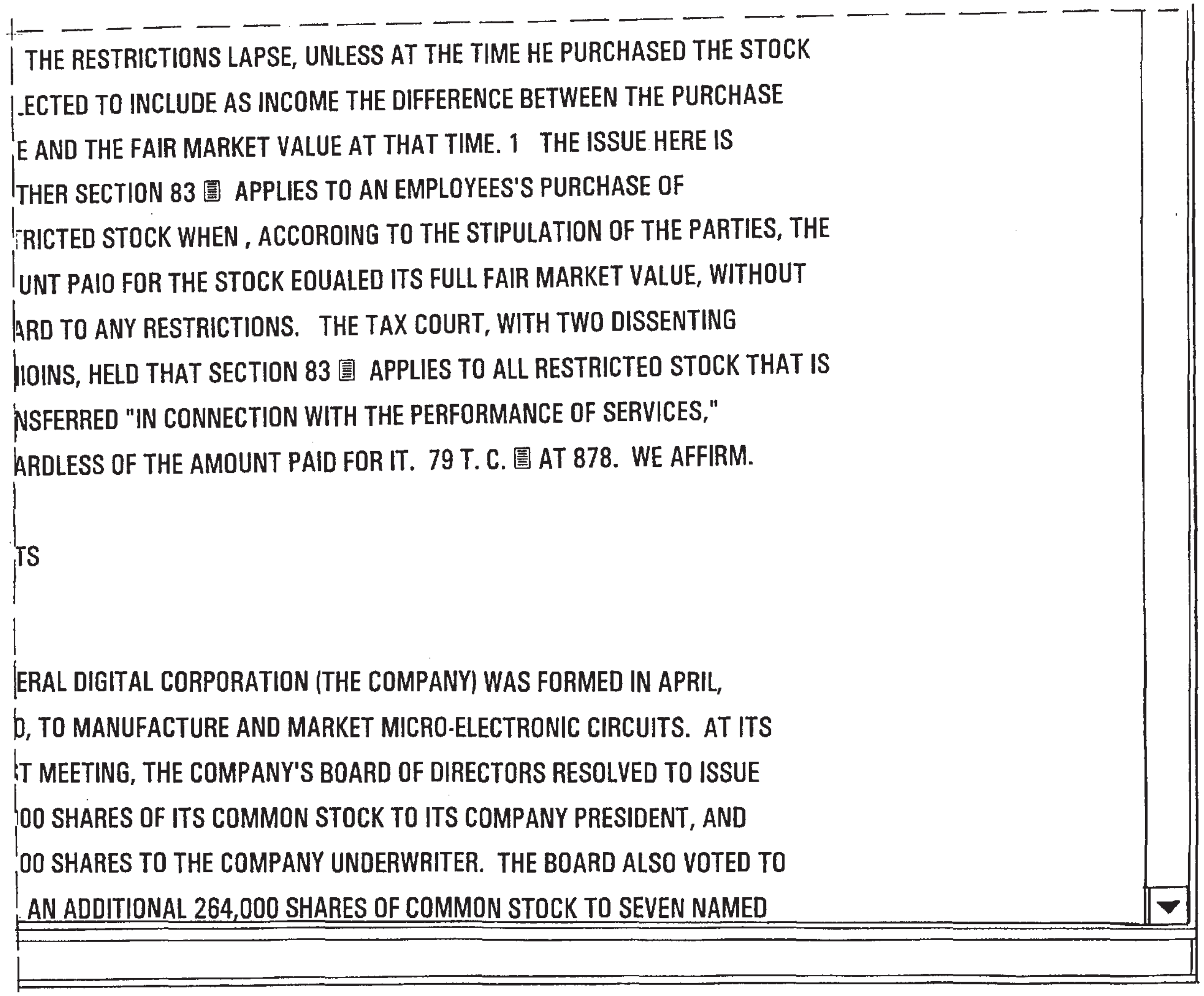


Fig. 8-3

Facebook Inc. Ex. 1001

Fig. 8-4



THE RESTRICTIONS LAPSE, UNLESS AT THE TIME HE PURCHASED THE STOCK
.ECTED TO INCLUDE AS INCOME THE DIFFERENCE BETWEEN THE PURCHASE
E AND THE FAIR MARKET VALUE AT THAT TIME. 1 THE ISSUE HERE IS
THER SECTION 83 APPLIES TO AN EMPLOYEES'S PURCHASE OF
RICTED STOCK WHEN , ACCOROING TO THE STIPULATION OF THE PARTIES, THE
UNT PAIO FOR THE STOCK EOUALED ITS FULL FAIR MARKET VALUE, WITHOUT
ARD TO ANY RESTRICTIONS. THE TAX COURT, WITH TWO DISSENTING
IIOINS, HELD THAT SECTION 83 APPLIES TO ALL RESTRICTEO STOCK THAT IS
NSFERRED "IN CONNECTION WITH THE PERFORMANCE OF SERVICES,"
ARDLESS OF THE AMOUNT PAID FOR IT. 79 T. C. AT 878. WE AFFIRM.
TS
ERAL DIGITAL CORPORATION (THE COMPANY) WAS FORMED IN APRIL,
D, TO MANUFACTURE AND MARKET MICRO-ELECTRONIC CIRCUITS. AT ITS
T MEETING, THE COMPANY'S BOARD OF DIRECTORS RESOLVED TO ISSUE
00 SHARES OF ITS COMMON STOCK TO ITS COMPANY PRESIDENT, AND
00 SHARES TO THE COMPANY UNDERWRITER. THE BOARD ALSO VOTED TO
AN ADDITIONAL 264,000 SHARES OF COMMON STOCK TO SEVEN NAMED

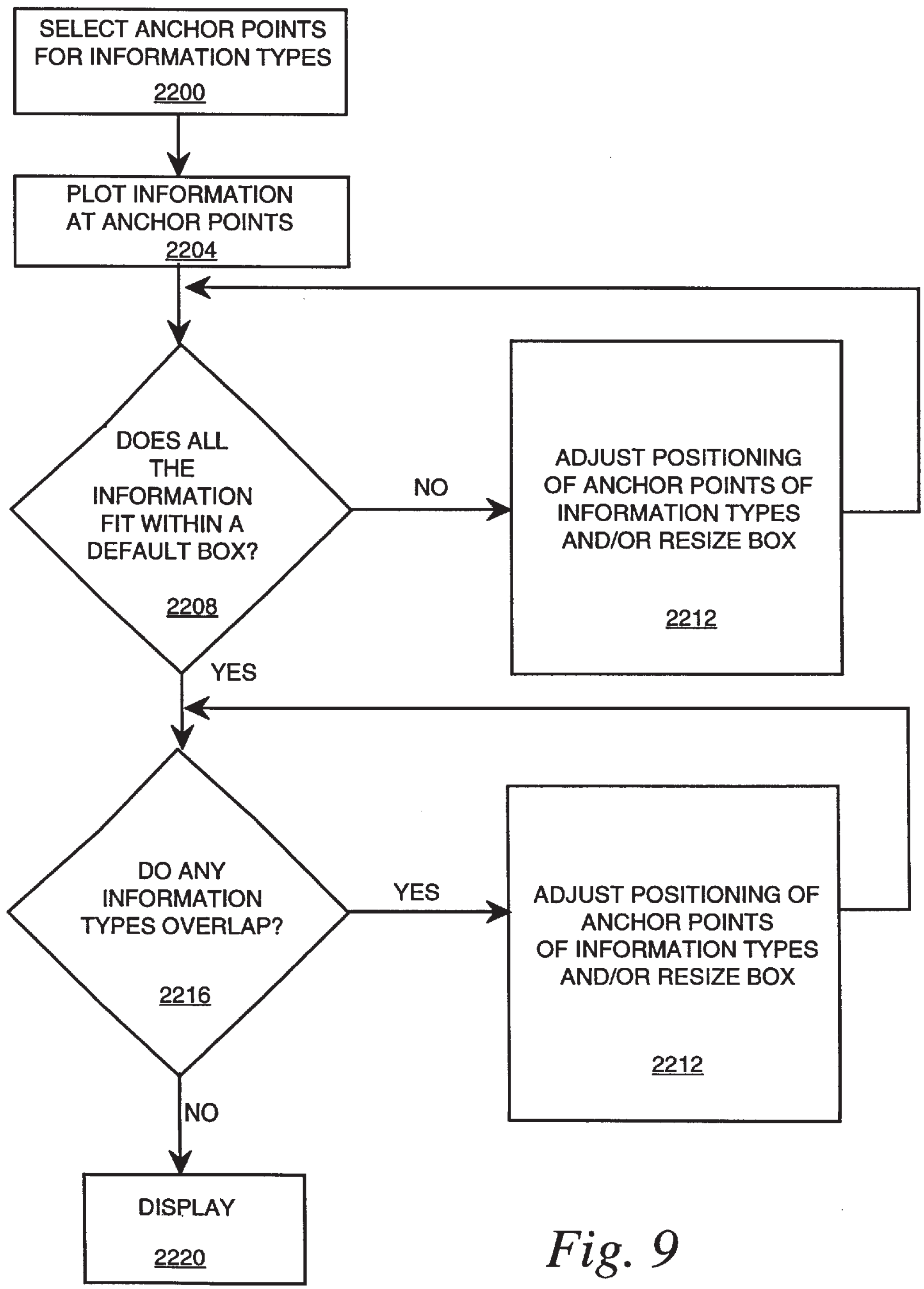

SELECT ANCHOR POINTS FOR INFORMATION TYPES
2200
PLOT INFORMATION AT ANCHOR POINTS
2204
DOES ALL THE INFORMATION FIT WITHIN A DEFAULT BOX?
2208
NO
ADJUST POSITIONING OF ANCHOR POINTS OF INFORMATION TYPES AND/OR RESIZE BOX
2212
YES
DO ANY INFORMATION TYPES OVERLAP?
2216
YES
ADJUST POSITIONING OF ANCHOR POINTS OF INFORMATION TYPES AND/OR RESIZE BOX
2212
NO
DISPLAY
2220

Fig. 9

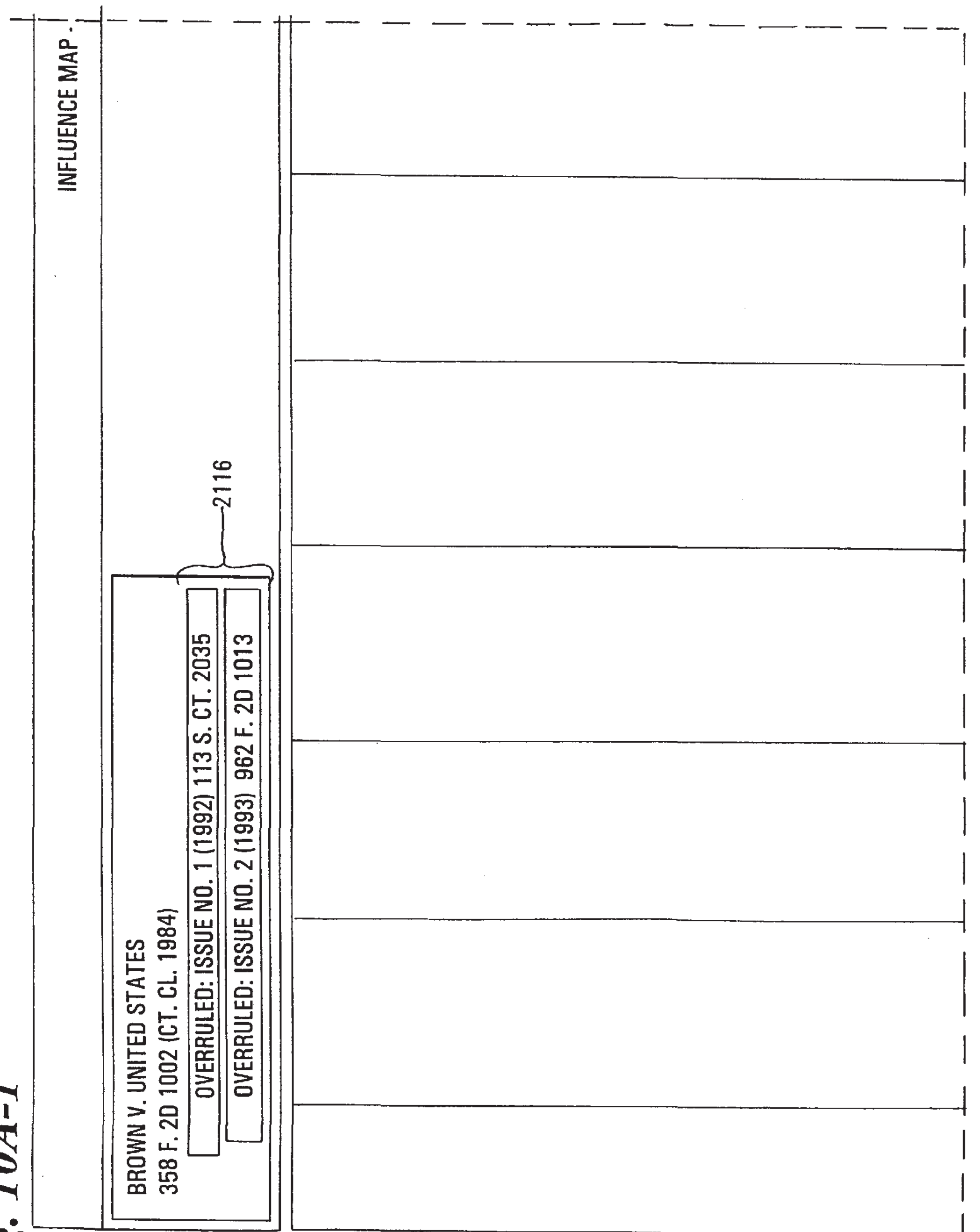
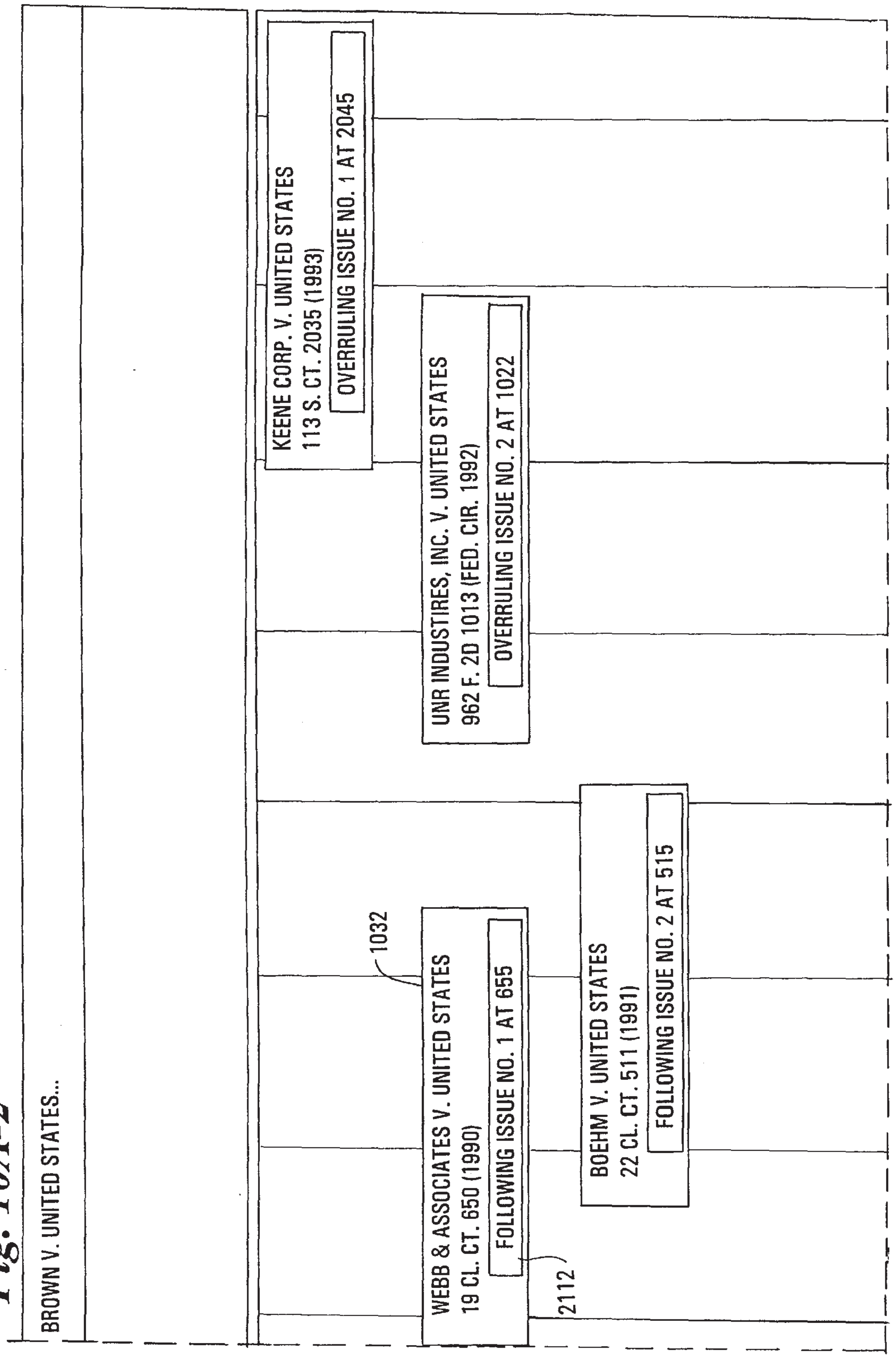

INFLUENCE MAP
BROWN V. UNITED STATES
358 F. 2D 1002 (CT. CL. 1984)
OVERRULED: ISSUE NO. 1 (1992) 113 S. CT. 2035
OVERRULED: ISSUE NO. 2 (1993) 962 F. 2D 1013
2116

Fig. 10A-1

Fig. 10A-2


BROWN V. UNITED STATES...
KEENE CORP. V. UNITED STATES
113 S. CT. 2035 (1993)
OVERRULING ISSUE NO. 1 AT 2045
UNR INDUSTIRES, INC. V. UNITED STATES
962 F. 2D 1013 (FED. CIR. 1992)
OVERRULING ISSUE NO. 2 AT 1022
1032
WEBB & ASSOCIATES V. UNITED STATES
19 CL. CT. 650 (1990)
FOLLOWING ISSUE NO. 1 AT 655
2112
BOEHM V. UNITED STATES
22 CL. CT. 511 (1991)
FOLLOWING ISSUE NO. 2 AT 515

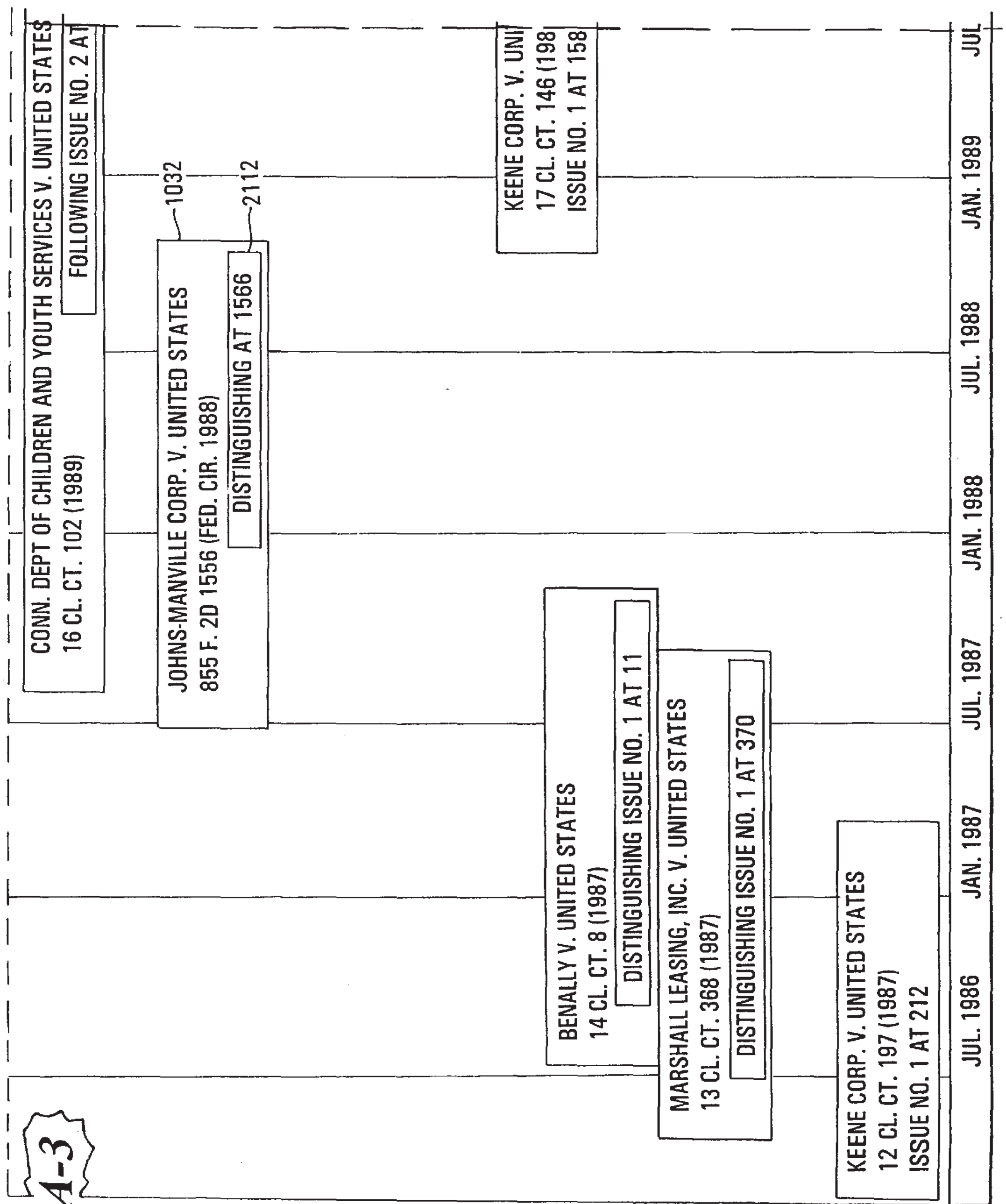

CONN. DEPT OF CHILDREN AND YOUTH SERVICES V. UNITED STATES
16 CL. CT. 102 (1989)
FOLLOWING ISSUE NO. 2 AT
1032
2112
JOHNS-MANVILLE CORP. V. UNITED STATES
855 F. 2D 1556 (FED. CIR. 1988)
DISTINGUISHING AT 1566
KEENE CORP. V. UNI
17 CL. CT. 146 (198
ISSUE NO. 1 AT 158
BENALLY V. UNITED STATES
14 CL. CT. 8 (1987)
DISTINGUISHING ISSUE NO. 1 AT 11
MARSHALL LEASING, INC. V. UNITED STATES
13 CL. CT. 368 (1987)
DISTINGUISHING ISSUE NO. 1 AT 370
KEENE CORP. V. UNITED STATES
12 CL. CT. 197 (1987)
ISSUE NO. 1 AT 212
JUL. 1986
JAN. 1987
JUL. 1987
JAN. 1988
JUL. 1988
JAN. 1989
JUL.

Fig. 10A-3

Fig. 10A-4

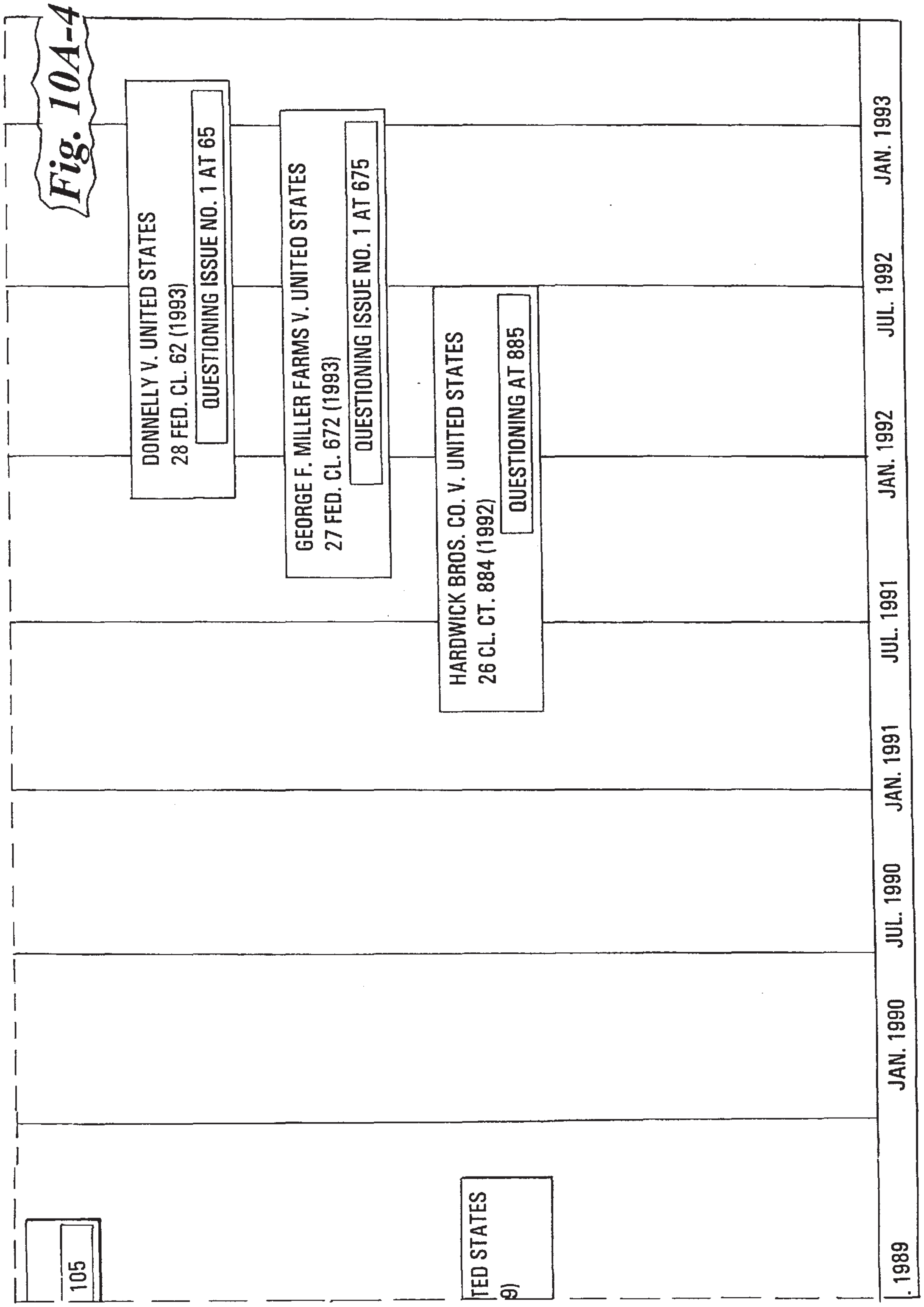

DONNELLY V. UNITED STATES
28 FED. CL. 62 (1993)
QUESTIONING ISSUE NO. 1 AT 65
GEORGE F. MILLER FARMS V. UNITED STATES
27 FED. CL. 672 (1993)
QUESTIONING ISSUE NO. 1 AT 675
HARDWICK BROS. CO. V. UNITED STATES
26 CL. CT. 884 (1992)
QUESTIONING AT 885
TED STATES
9)
105
. 1989
JAN. 1990
JUL. 1990
JAN. 1991
JUL. 1991
JAN. 1992
JUL. 1992
JAN. 1993

Fig.10B-1

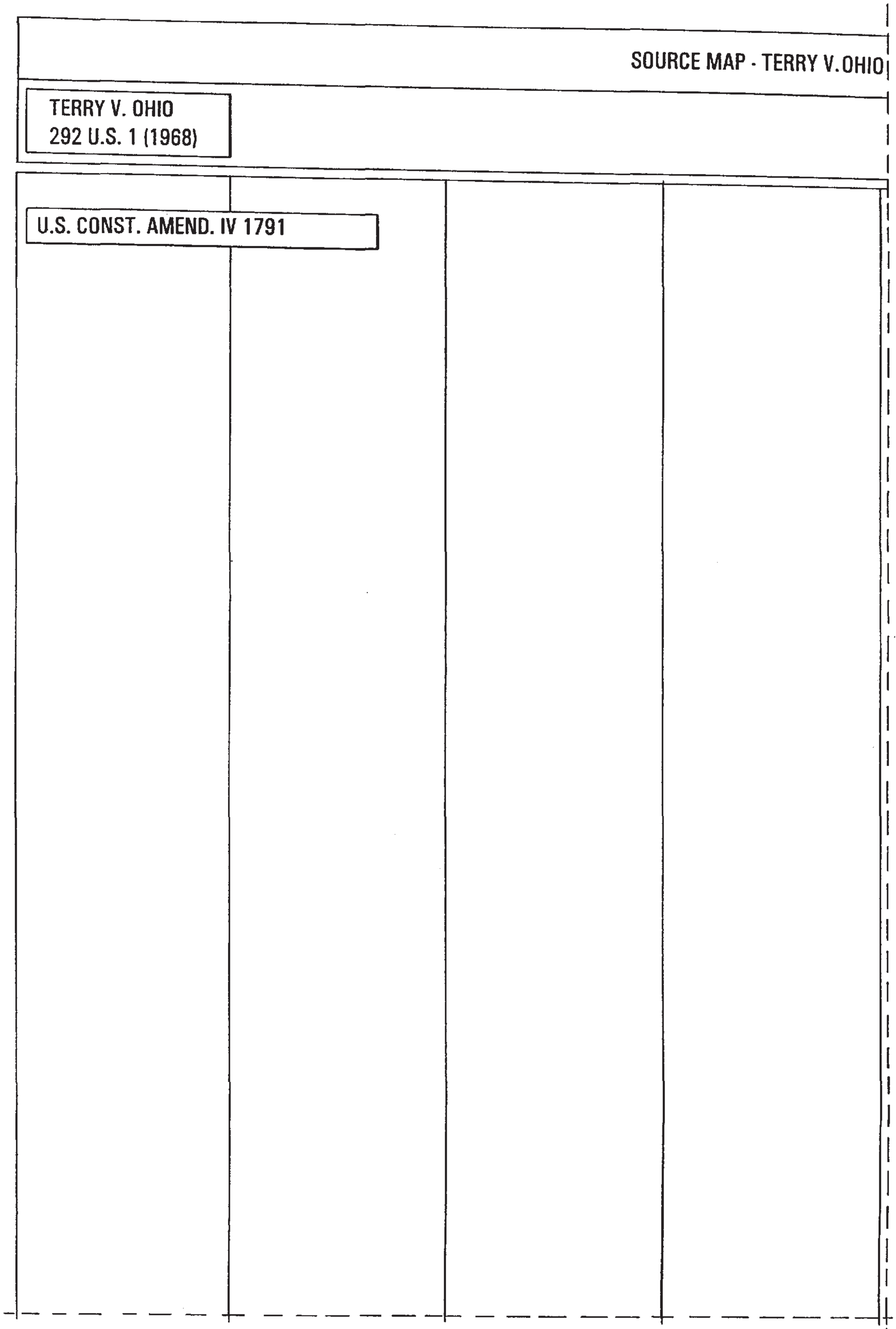
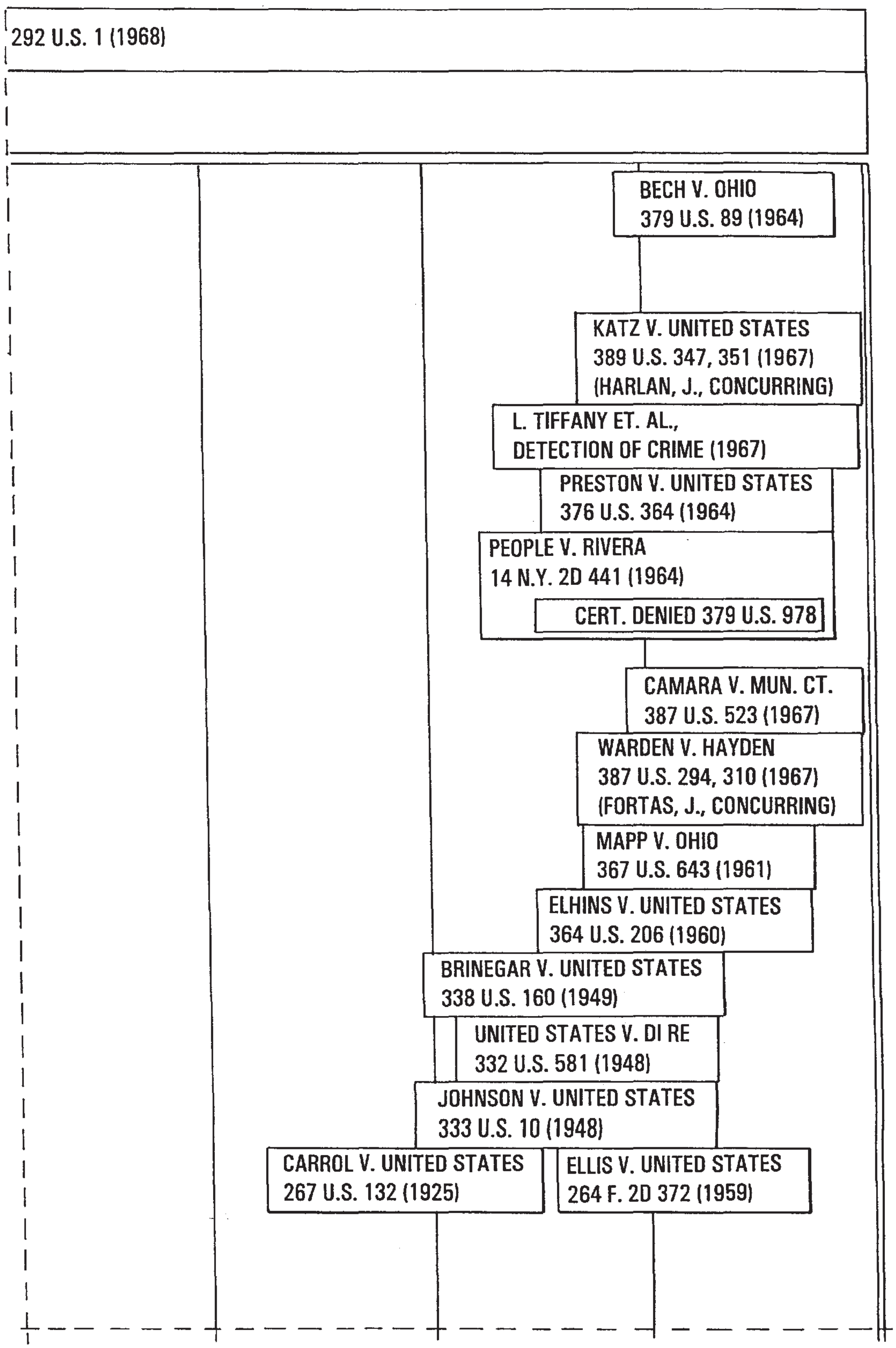

SOURCE MAP - TERRY V. OHIO
TERRY V. OHIO
292 U.S. 1 (1968)
U.S. CONST. AMEND. IV 1791

Fig.10B-2



292 U.S. 1 (1968)
BECH V. OHIO
379 U.S. 89 (1964)
KATZ V. UNITED STATES
389 U.S. 347, 351 (1967)
(HARLAN, J., CONCURRING)
L. TIFFANY ET. AL.,
DETECTION OF CRIME (1967)
PRESTON V. UNITED STATES
376 U.S. 364 (1964)
PEOPLE V. RIVERA
14 N.Y. 2D 441 (1964)
CERT. DENIED 379 U.S. 978
CAMARA V. MUN. CT.
387 U.S. 523 (1967)
WARDEN V. HAYDEN
387 U.S. 294, 310 (1967)
(FORTAS, J., CONCURRING)
MAPP V. OHIO
367 U.S. 643 (1961)
ELHINS V. UNITED STATES
364 U.S. 206 (1960)
BRINEGAR V. UNITED STATES
338 U.S. 160 (1949)
UNITED STATES V. DI RE
332 U.S. 581 (1948)
JOHNSON V. UNITED STATES
333 U.S. 10 (1948)
CARROL V. UNITED STATES
267 U.S. 132 (1925)
ELLIS V. UNITED STATES
264 F. 2D 372 (1959)

Fig.10B-3

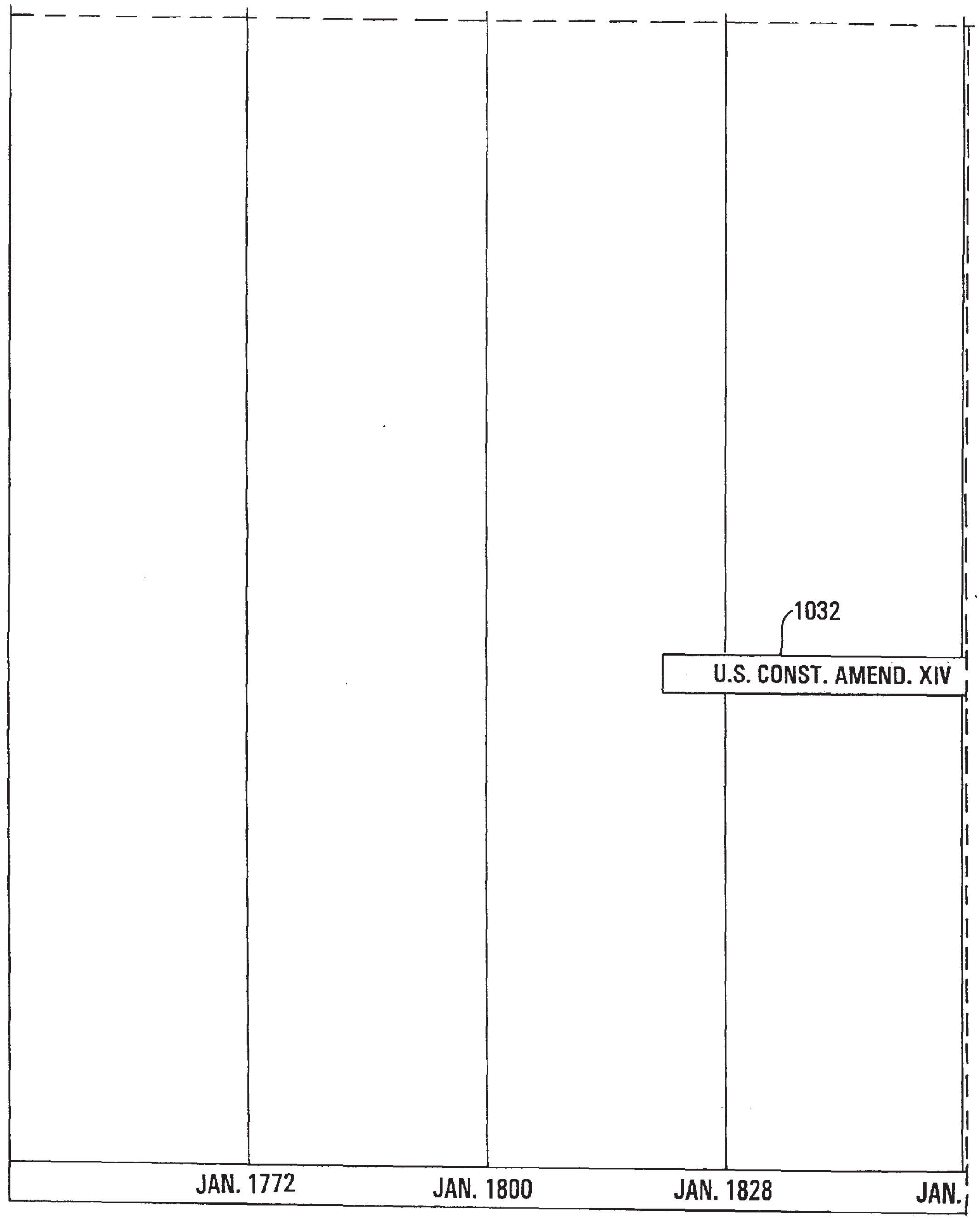
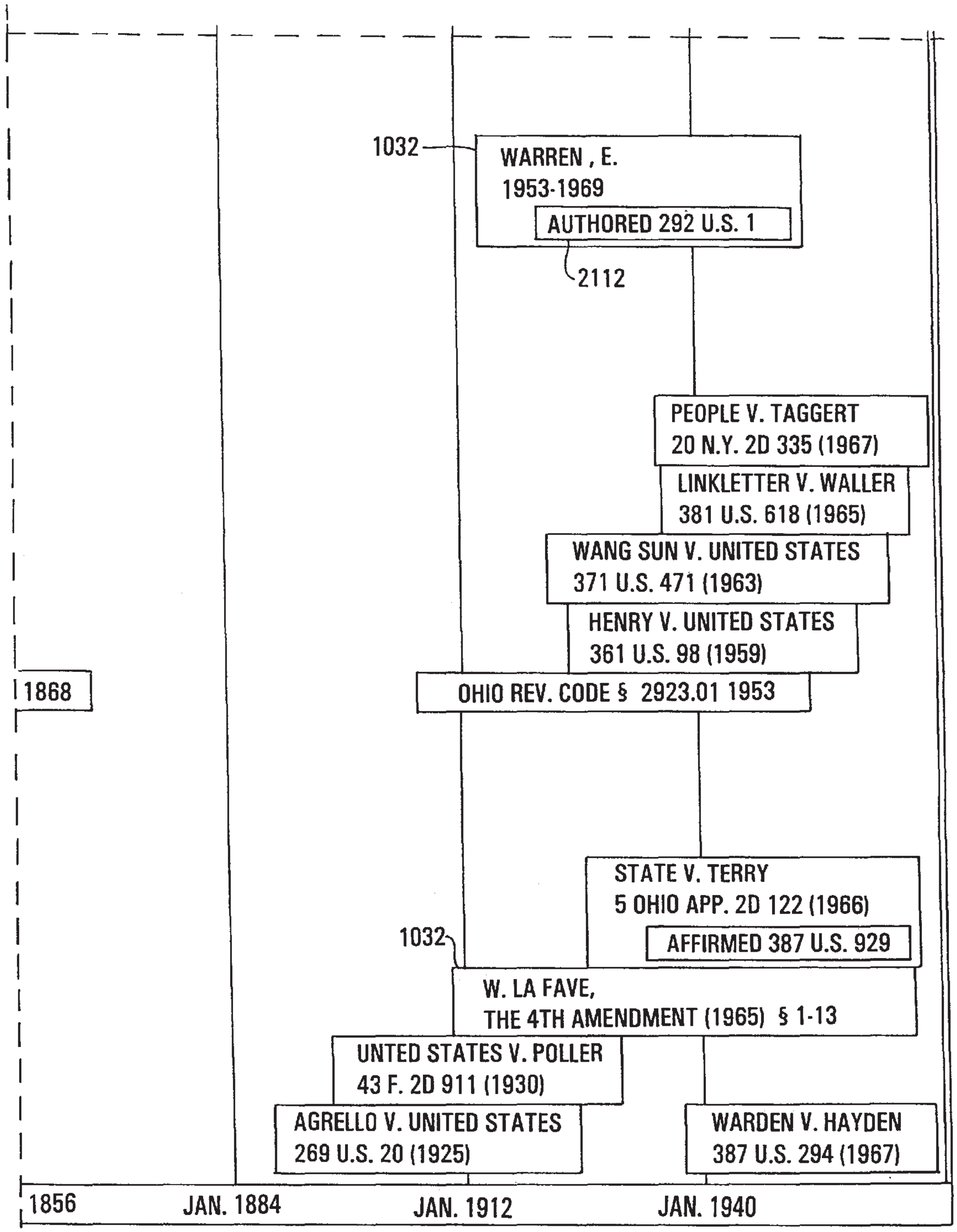


Fig.10B-4


1032
WARREN, E.
1953-1969
AUTHORED 292 U.S. 1
2112
PEOPLE V. TAGGERT
20 N.Y. 2D 335 (1967)
LINKLETTER V. WALLER
381 U.S. 618 (1965)
WANG SUN V. UNITED STATES
371 U.S. 471 (1963)
HENRY V. UNITED STATES
361 U.S. 98 (1959)
1868
OHIO REV. CODE § 2923.01 1953
STATE V. TERRY
5 OHIO APP. 2D 122 (1966)
AFFIRMED 387 U.S. 929
1032
W. LA FAVE,
THE 4TH AMENDMENT (1965) § 1-13
UNTED STATES V. POLLER
43 F. 2D 911 (1930)
AGRELLO V. UNITED STATES
269 U.S. 20 (1925)
WARDEN V. HAYDEN
387 U.S. 294 (1967)
1856
JAN. 1884
JAN. 1912
JAN. 1940

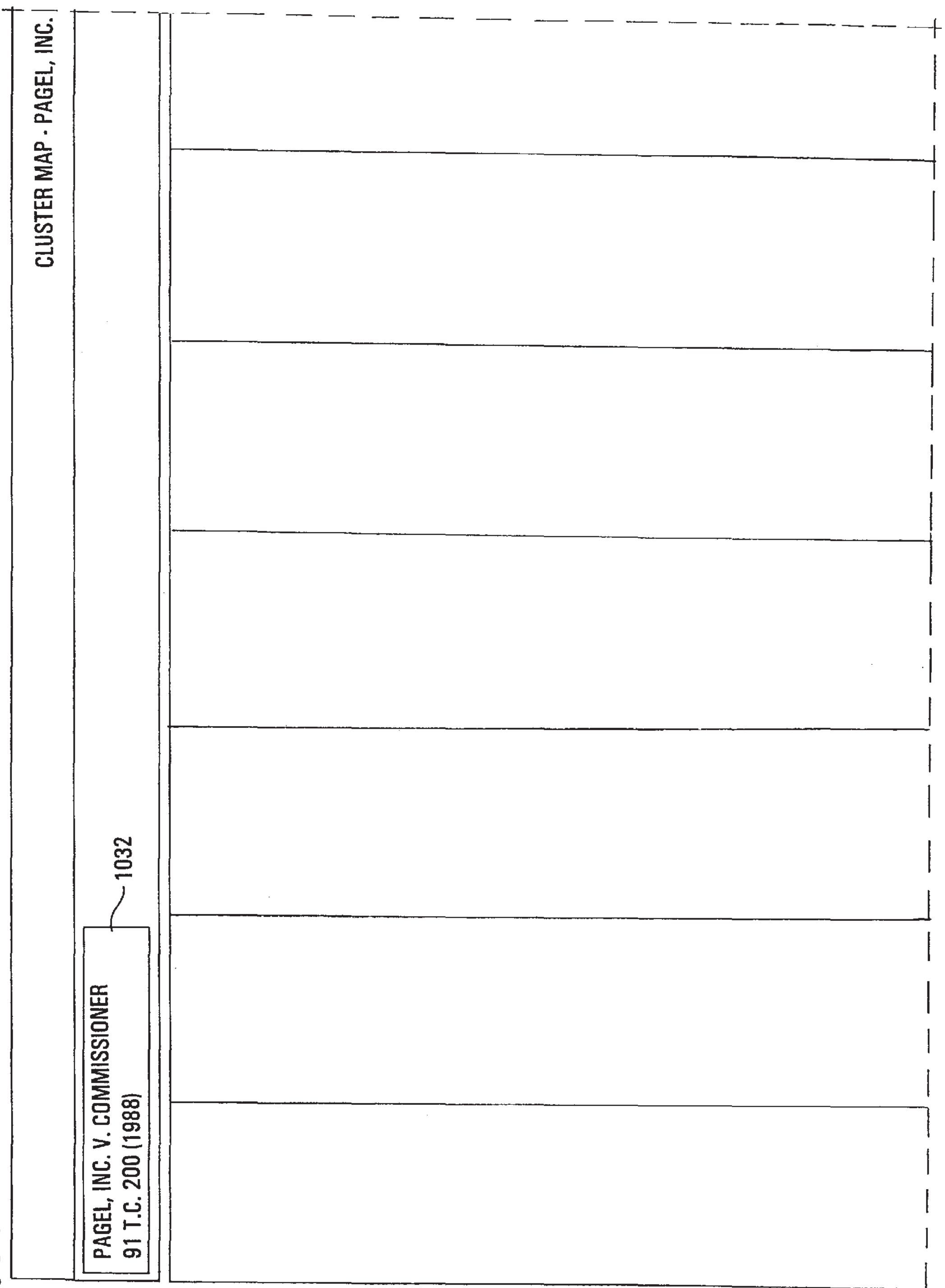
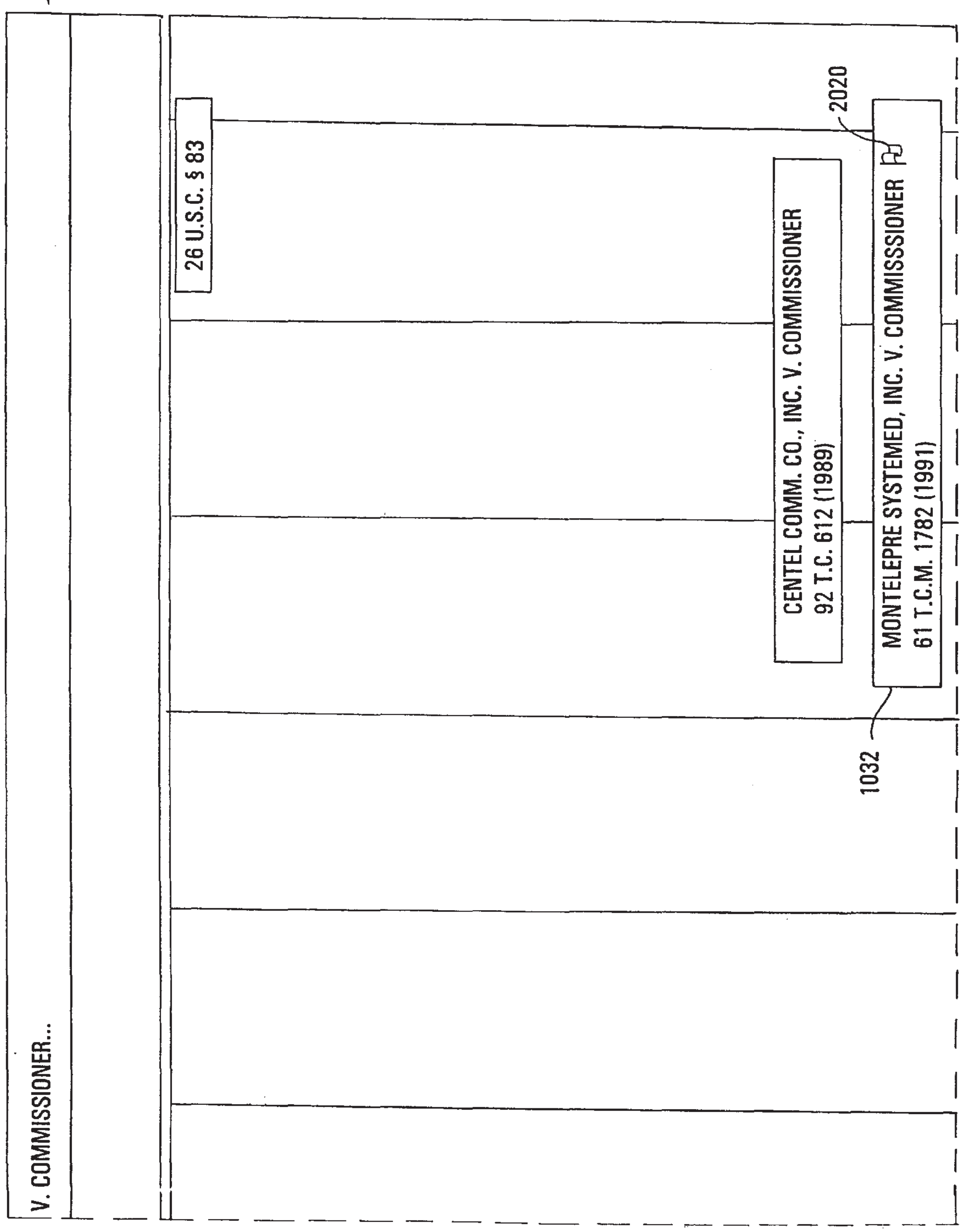

CLUSTER MAP - PAGEL, INC.
1032
PAGEL, INC. V. COMMISSIONER
91 T.C. 200 (1988)

Fig. 10C-1

Fig. 10C-2


V. COMMISSIONER...
26 U.S.C. § 83
CENTEL COMM. CO., INC. V. COMMISSIONER
92 T.C. 612 (1989)
2020
MONTELEPRE SYSTEMED, INC. V. COMMISSSIONER
61 T.C.M. 1782 (1991)
1032

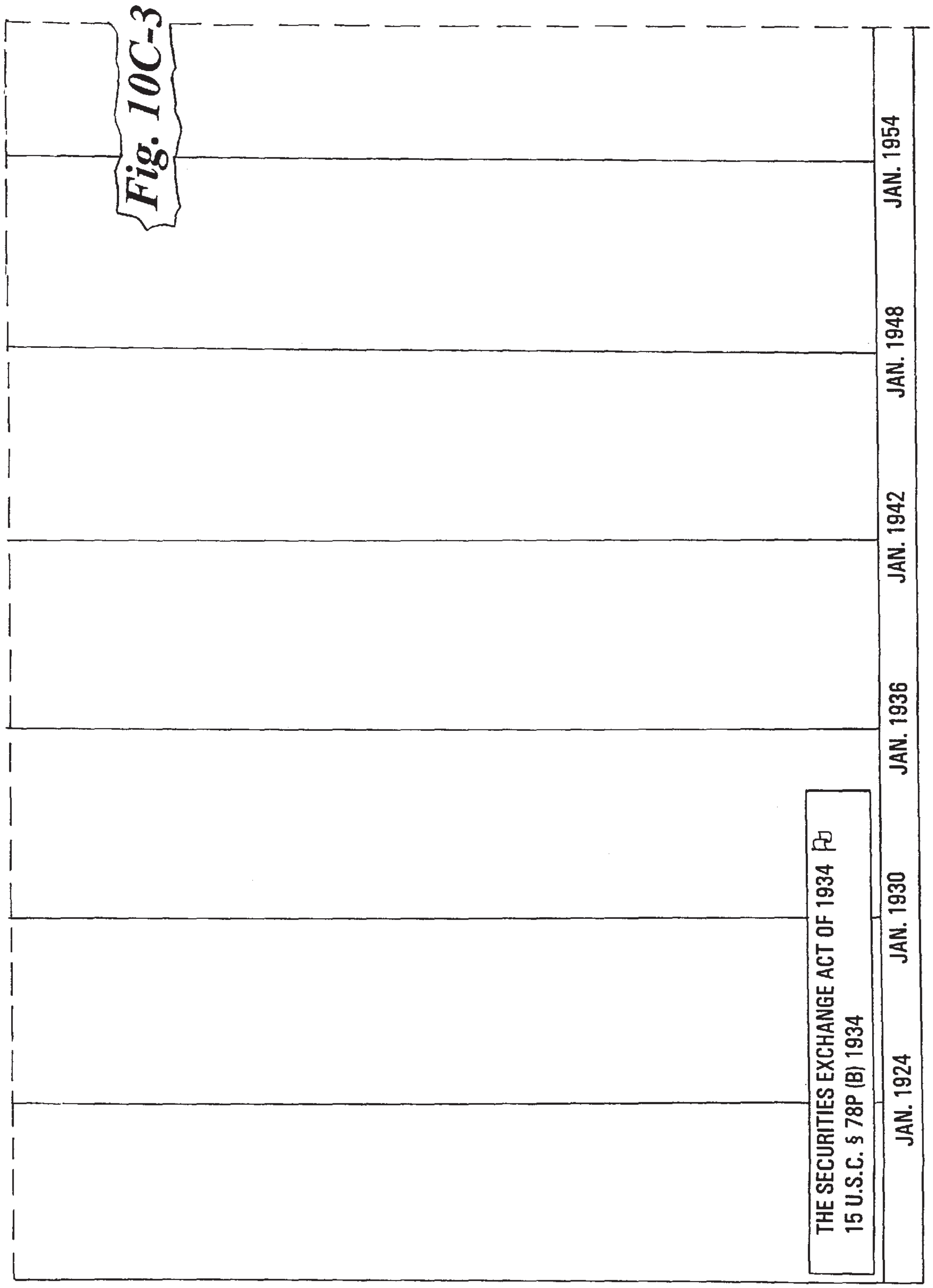
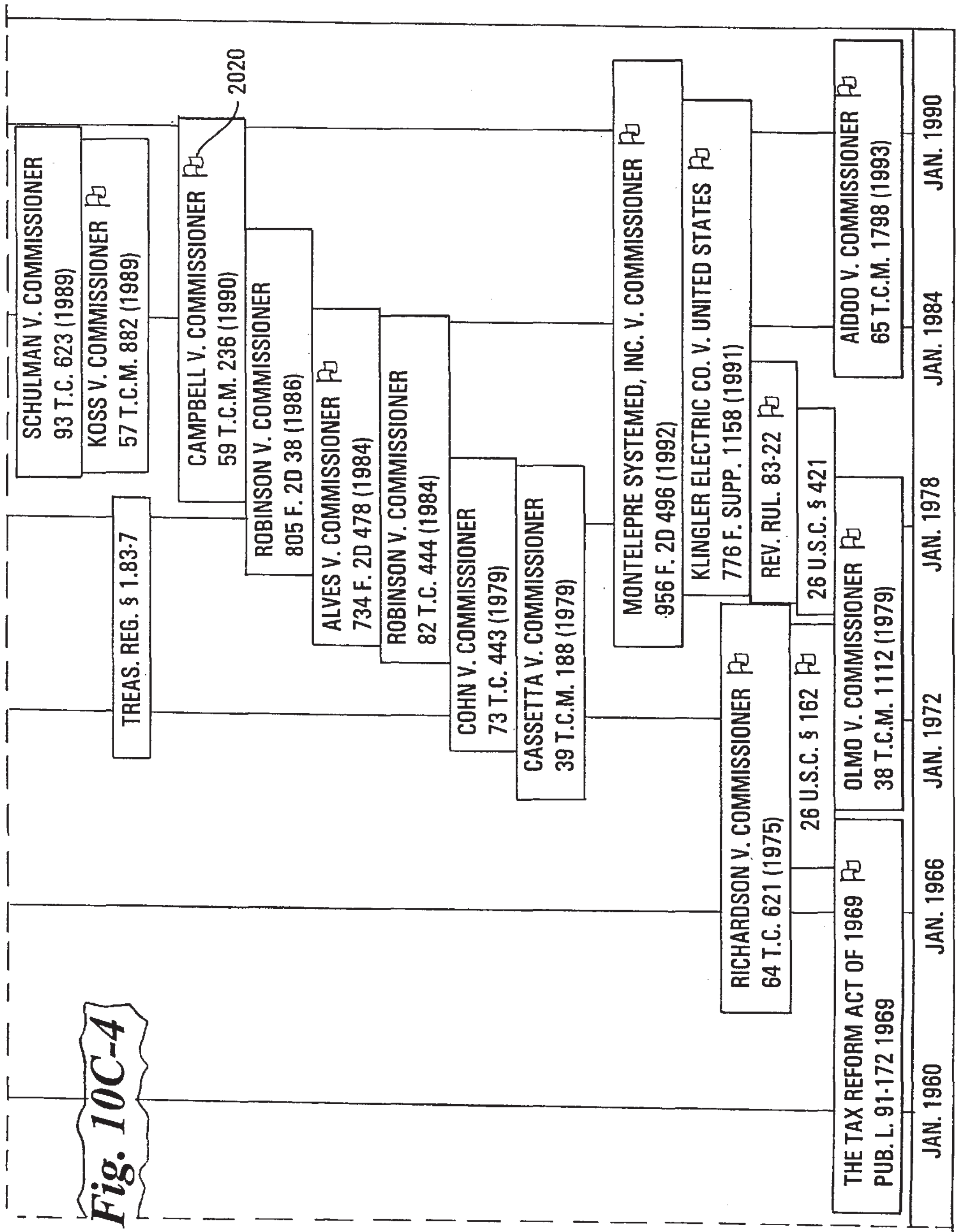
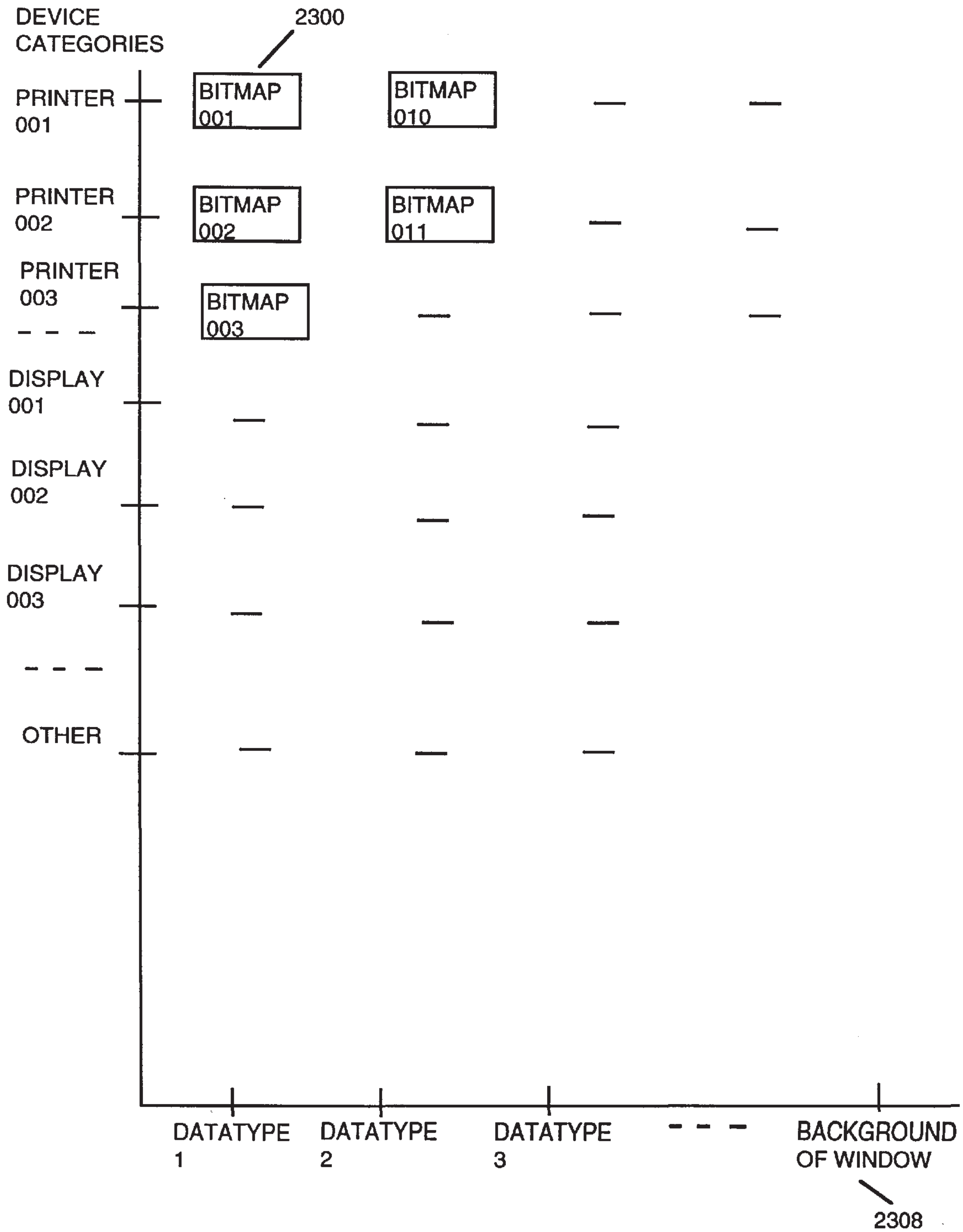

JAN. 1954
JAN. 1948
JAN. 1942
JAN. 1936
JAN. 1930
JAN. 1924
THE SECURITIES EXCHANGE ACT OF 1934
15 U.S.C. § 78P (B) 1934

Fig. 10C-3



2020
SCHULMAN V. COMMISSIONER
93 T.C. 623 (1989)
KOSS V. COMMISSIONER
57 T.C.M. 882 (1989)
CAMPBELL V. COMMISSIONER
59 T.C.M. 236 (1990)
ROBINSON V. COMMISSIONER
805 F. 2D 38 (1986)
ALVES V. COMMISSIONER
734 F. 2D 478 (1984)
ROBINSON V. COMMISSIONER
82 T.C. 444 (1984)
COHN V. COMMISSIONER
73 T.C. 443 (1979)
CASSETTA V. COMMISSIONER
39 T.C.M. 188 (1979)
TREAS. REG. § 1.83-7
MONTELEPRE SYSTEMED, INC. V. COMMISSIONER
956 F. 2D 496 (1992)
KLINGLER ELECTRIC CO. V. UNITED STATES
776 F. SUPP. 1158 (1991)
REV. RUL. 83-22
26 U.S.C. § 421
RICHARDSON V. COMMISSIONER
64 T.C. 621 (1975)
26 U.S.C. § 162
AIDOO V. COMMISSIONER
65 T.C.M. 1798 (1993)
OLMO V. COMMISSIONER
38 T.C.M. 1112 (1979)
THE TAX REFORM ACT OF 1969
PUB. L. 91-172 1969
JAN. 1960
JAN. 1966
JAN. 1972
JAN. 1978
JAN. 1984
JAN. 1990

Fig. 10C-4



DEVICE CATEGORIES
2300
PRINTER 001
BITMAP 001
BITMAP 010
PRINTER 002
BITMAP 002
BITMAP 011
PRINTER 003
BITMAP 003
DISPLAY 001
DISPLAY 002
DISPLAY 003
OTHER
DATATYPE 1
DATATYPE 2
DATATYPE 3
BACKGROUND OF WINDOW
2308

Fig. 11

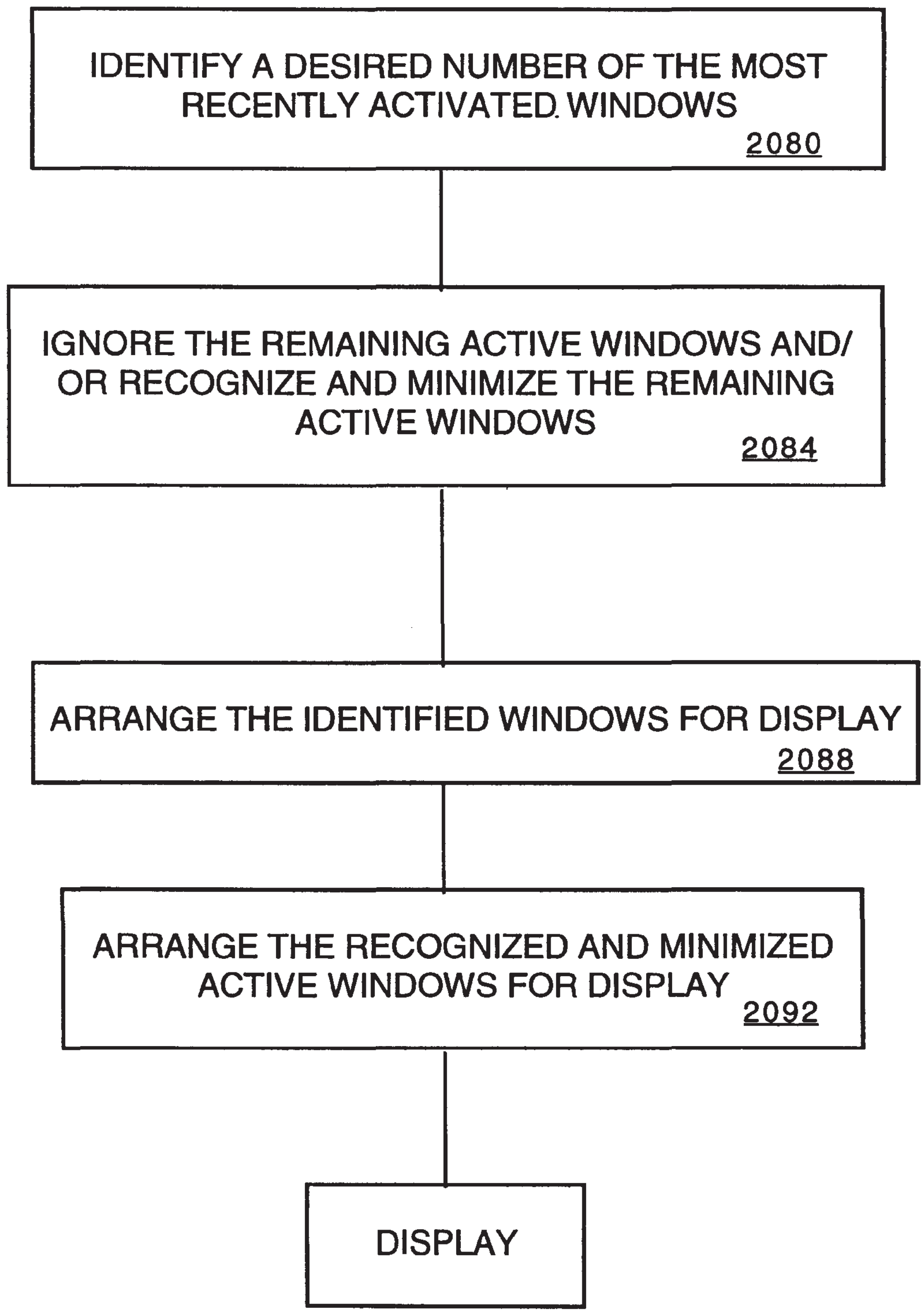
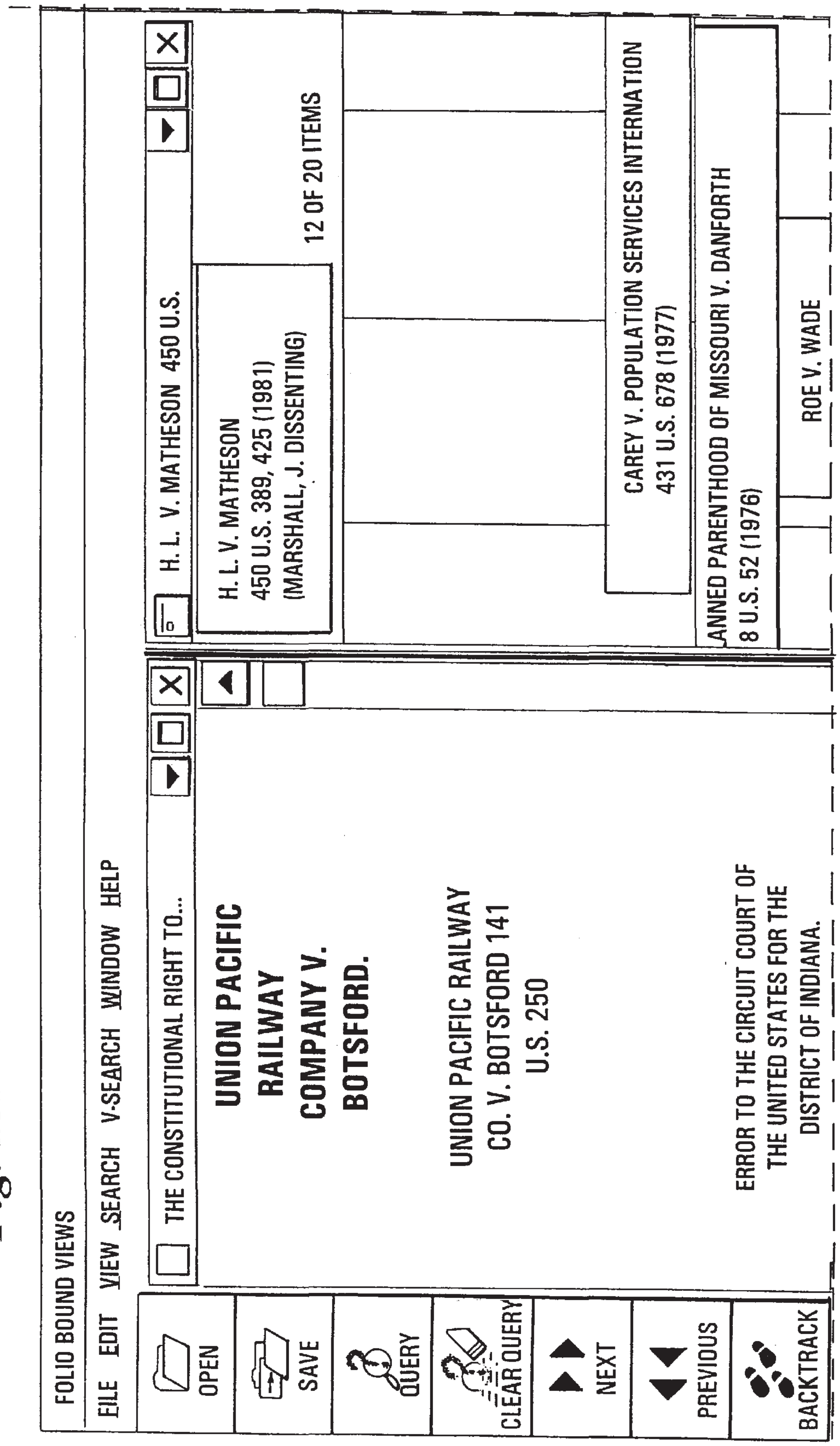
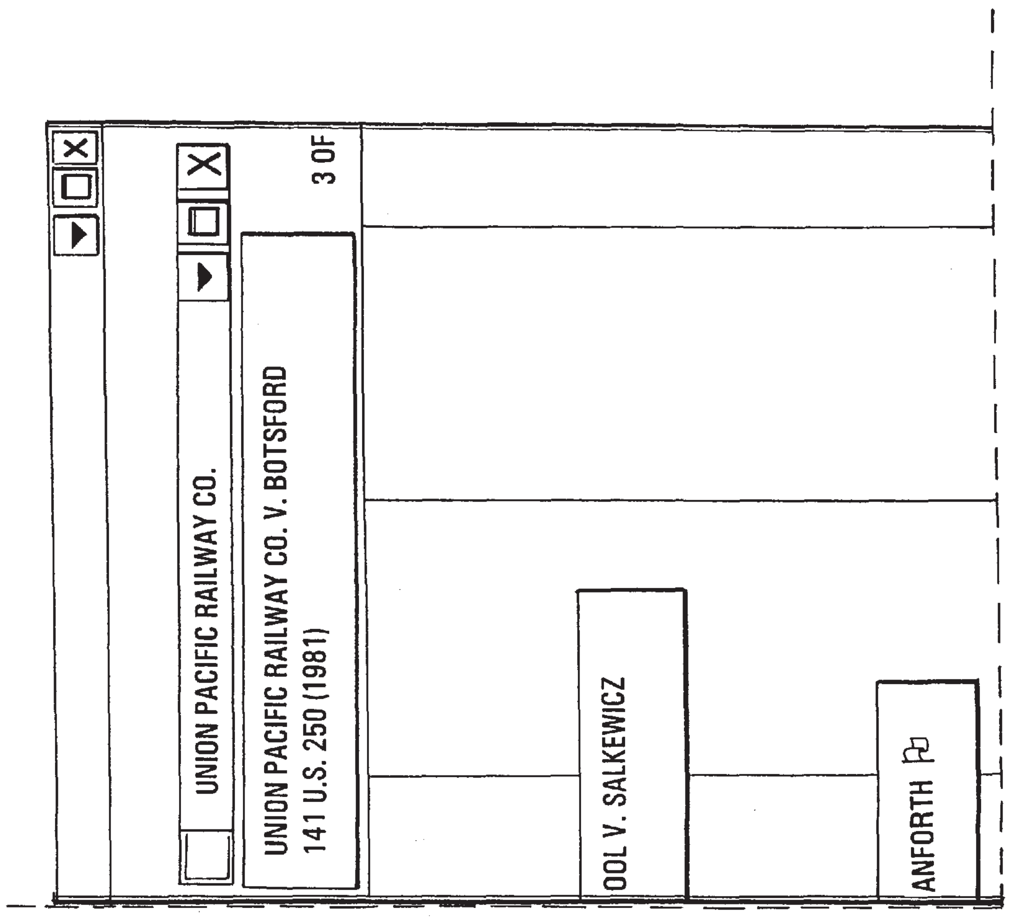

IDENTIFY A DESIRED NUMBER OF THE MOST RECENTLY ACTIVATED WINDOWS
2080
IGNORE THE REMAINING ACTIVE WINDOWS AND/ OR RECOGNIZE AND MINIMIZE THE REMAINING ACTIVE WINDOWS
2084
ARRANGE THE IDENTIFIED WINDOWS FOR DISPLAY
2088
ARRANGE THE RECOGNIZED AND MINIMIZED ACTIVE WINDOWS FOR DISPLAY
2092
DISPLAY

Fig. 12

Fig. 13A-1

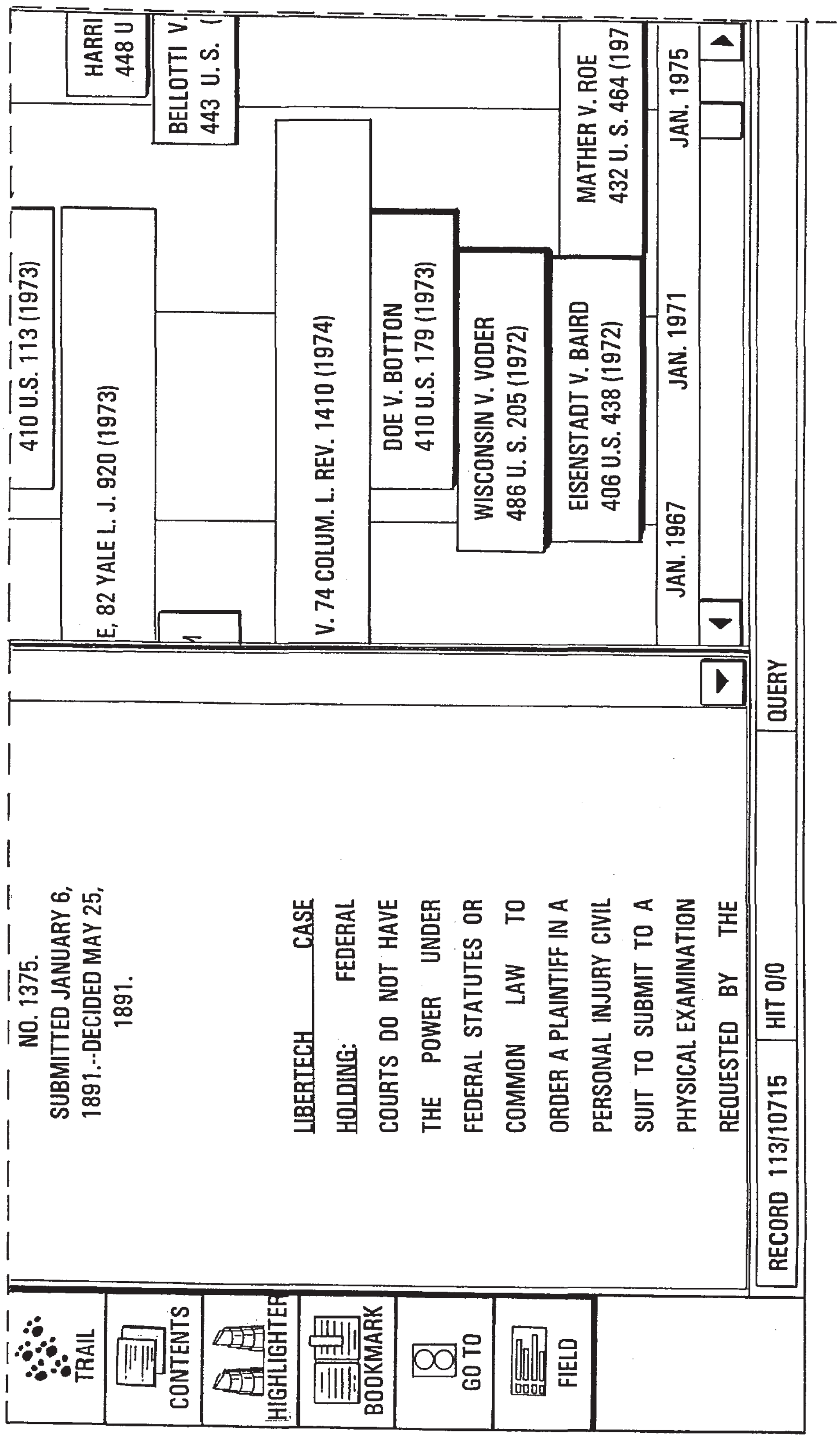

FOLIO BOUND VIEWS
FILE EDIT VIEW SEARCH V-SEARCH WINDOW HELP
OPEN
SAVE
QUERY
CLEAR QUERY
NEXT
PREVIOUS
BACKTRACK
THE CONSTITUTIONAL RIGHT TO...
UNION PACIFIC RAILWAY COMPANY V. BOTSFORD.
UNION PACIFIC RAILWAY CO. V. BOTSFORD 141 U.S. 250
ERROR TO THE CIRCUIT COURT OF THE UNITED STATES FOR THE DISTRICT OF INDIANA.
H. L. V. MATHESON 450 U.S.
H. L. V. MATHESON
450 U.S. 389, 425 (1981)
(MARSHALL, J. DISSENTING)
12 OF 20 ITEMS
CAREY V. POPULATION SERVICES INTERNATION
431 U.S. 678 (1977)
ANNED PARENTHOOD OF MISSOURI V. DANFORTH
8 U.S. 52 (1976)
ROE V. WADE



3 OF
UNION PACIFIC RAILWAY CO.
UNION PACIFIC RAILWAY CO. V. BOTSFORD
141 U.S. 250 (1981)
OOL V. SALKEWICZ
ANFORTH

Fig. 13A-2

Fig. 13A-3



TRAIL
CONTENTS
HIGHLIGHTER
BOOKMARK
GO TO
FIELD
NO. 1375.
SUBMITTED JANUARY 6, 1891.--DECIDED MAY 25, 1891.
LIBERTECH CASE
HOLDING: FEDERAL COURTS DO NOT HAVE THE POWER UNDER FEDERAL STATUTES OR COMMON LAW TO ORDER A PLAINTIFF IN A PERSONAL INJURY CIVIL SUIT TO SUBMIT TO A PHYSICAL EXAMINATION REQUESTED BY THE
RECORD 113/10715
HIT 0/0
QUERY
410 U.S. 113 (1973)
E, 82 YALE L. J. 920 (1973)
V. 74 COLUM. L. REV. 1410 (1974)
DOE V. BOTTON
410 U.S. 179 (1973)
WISCONSIN V. VODER
486 U. S. 205 (1972)
EISENSTADT V. BAIRD
406 U.S. 438 (1972)
MATHER V. ROE
432 U. S. 464 (197
BELLOTTI V.
443 U. S. (
HARRI
448 U
JAN. 1967
JAN. 1971
JAN. 1975

Fig. 13A-4

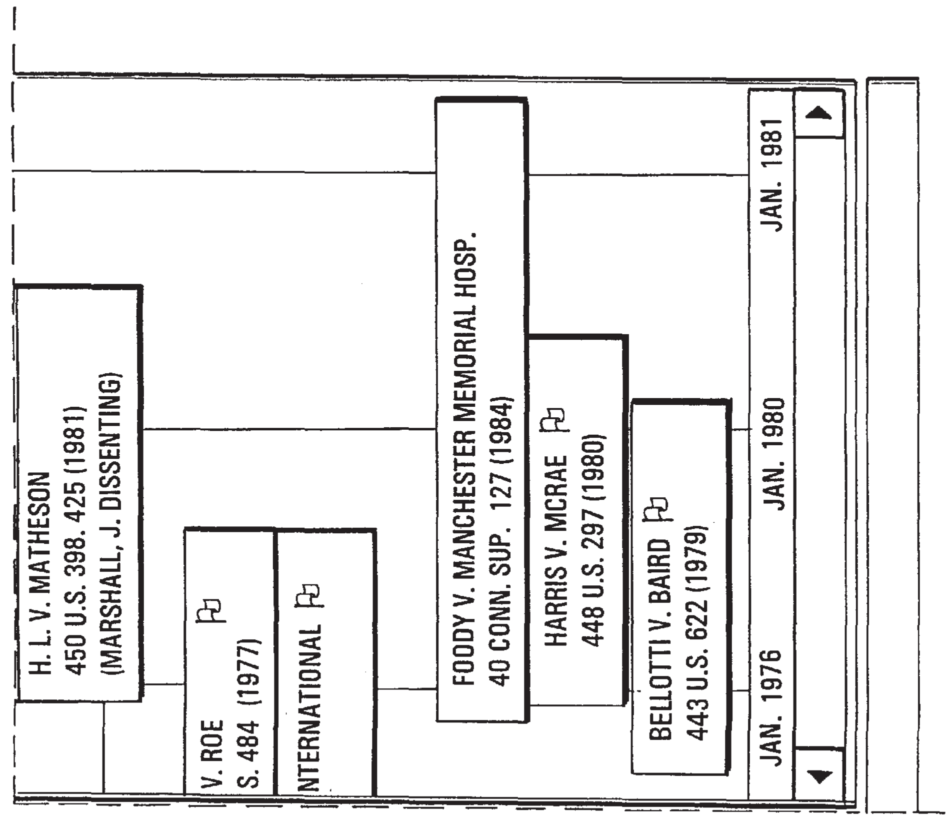
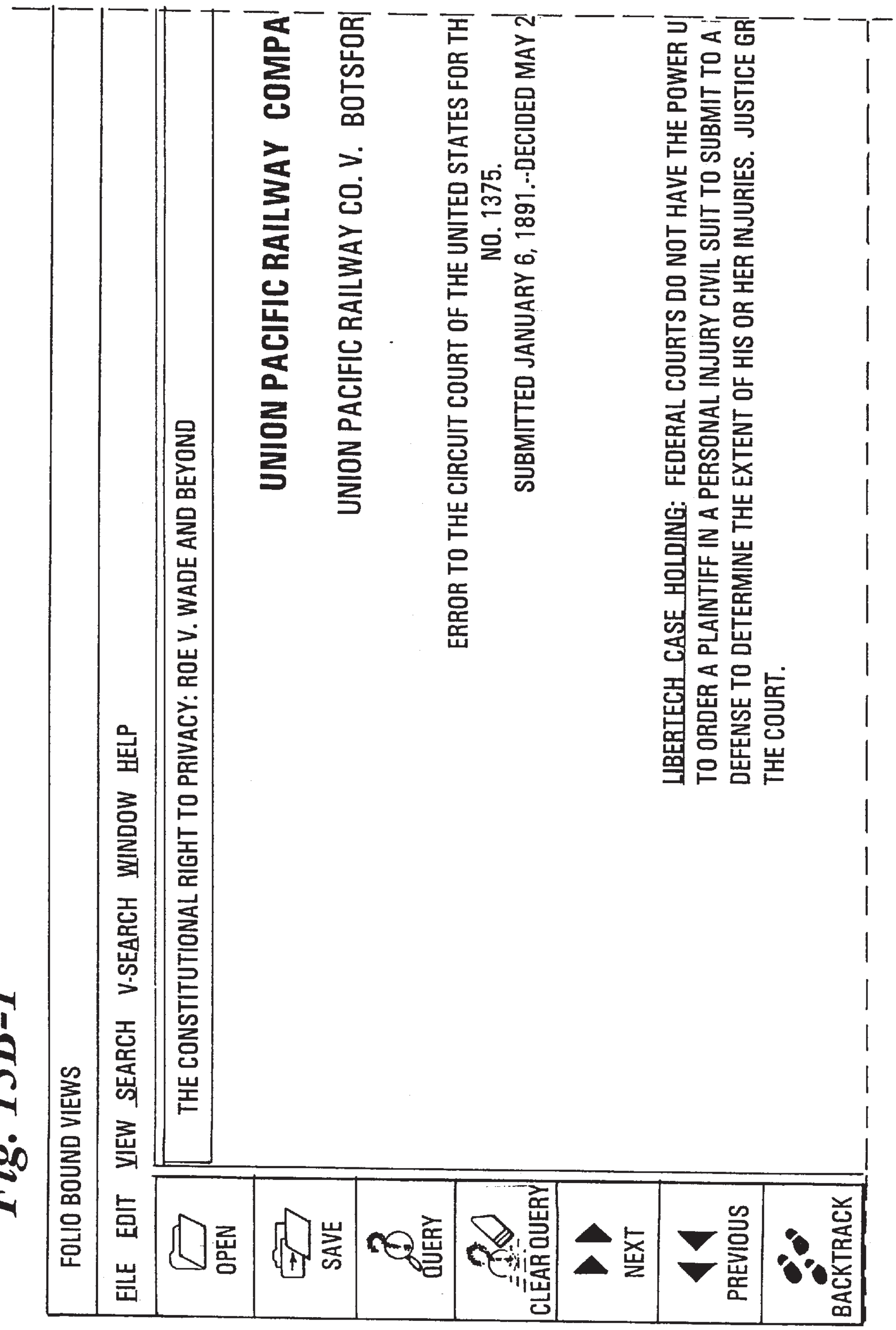

H. L. V. MATHESON
450 U.S. 398. 425 (1981)
(MARSHALL, J. DISSENTING)
V. ROE
S. 484 (1977)
NTERNATIONAL
FOODY V. MANCHESTER MEMORIAL HOSP.
40 CONN. SUP. 127 (1984)
HARRIS V. MCRAE
448 U.S. 297 (1980)
BELLOTTI V. BAIRD
443 U.S. 622 (1979)
JAN. 1976
JAN. 1980
JAN. 1981

Fig. 13B-1


FOLIO BOUND VIEWS
FILE EDIT VIEW SEARCH V-SEARCH WINDOW HELP
OPEN
SAVE
QUERY
CLEAR QUERY
NEXT
PREVIOUS
BACKTRACK
THE CONSTITUTIONAL RIGHT TO PRIVACY: ROE V. WADE AND BEYOND
UNION PACIFIC RAILWAY COMPA
UNION PACIFIC RAILWAY CO. V. BOTSFOR
ERROR TO THE CIRCUIT COURT OF THE UNITED STATES FOR TH
NO. 1375.
SUBMITTED JANUARY 6, 1891.--DECIDED MAY 2
LIBERTECH CASE HOLDING: FEDERAL COURTS DO NOT HAVE THE POWER U
TO ORDER A PLAINTIFF IN A PERSONAL INJURY CIVIL SUIT TO SUBMIT TO A
DEFENSE TO DETERMINE THE EXTENT OF HIS OR HER INJURIES. JUSTICE GR
THE COURT.

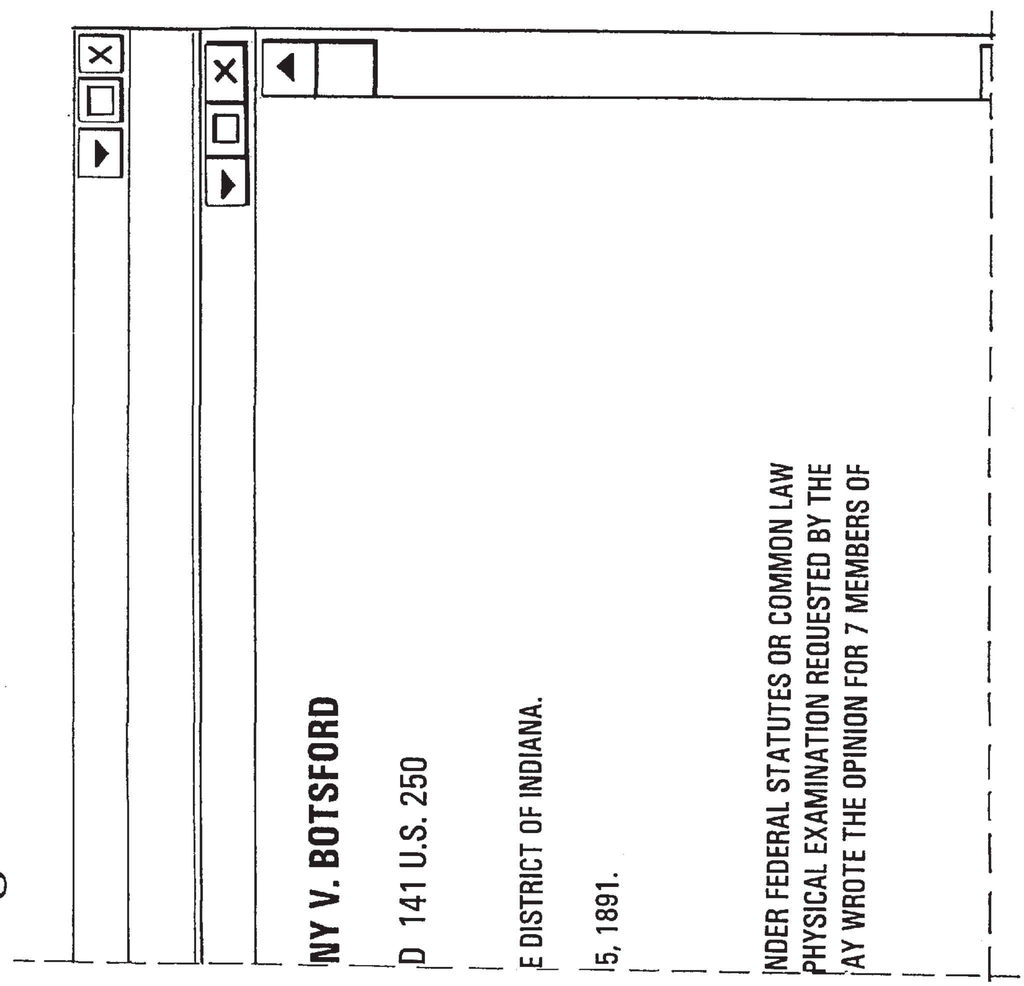

NY V. BOTSFORD
D 141 U.S. 250
E DISTRICT OF INDIANA.
5, 1891.
NDER FEDERAL STATUTES OR COMMON LAW
PHYSICAL EXAMINATION REQUESTED BY THE
AY WROTE THE OPINION FOR 7 MEMBERS OF

Fig. 13B-2

Fig. 13B-3

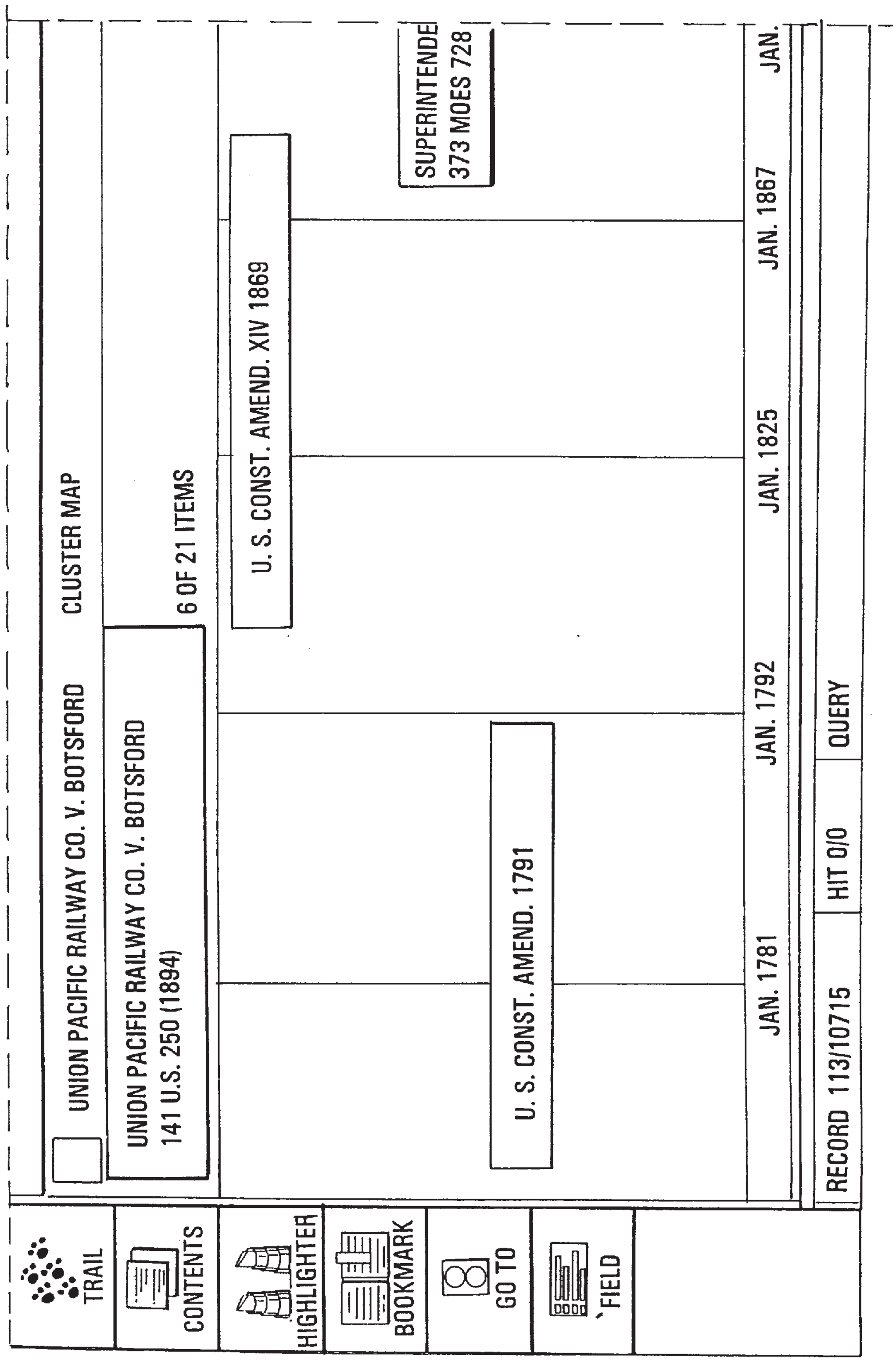
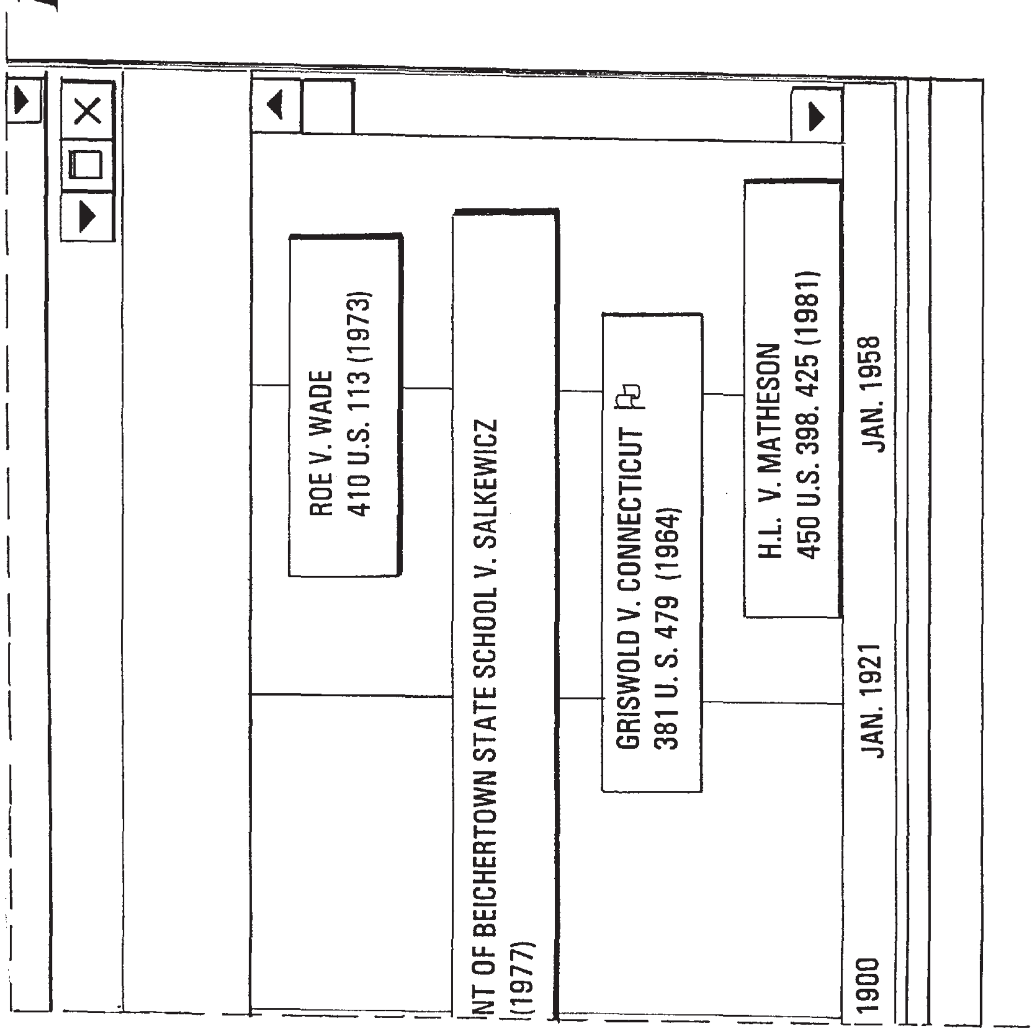

UNION PACIFIC RAILWAY CO. V. BOTSFORD
CLUSTER MAP
UNION PACIFIC RAILWAY CO. V. BOTSFORD
141 U.S. 250 (1894)
6 OF 21 ITEMS
U. S. CONST. AMEND. XIV 1869
SUPERINTENDE
373 MOES 728
U. S. CONST. AMEND. 1791
JAN. 1781
JAN. 1792
JAN. 1825
JAN. 1867
JAN.
RECORD 113/10715
HIT 0/0
QUERY
TRAIL
CONTENTS
HIGHLIGHTER
BOOKMARK
GO TO
FIELD

Fig. 13B-4


ROE V. WADE
410 U.S. 113 (1973)
NT OF BEICHERTOWN STATE SCHOOL V. SALKEWICZ
(1977)
GRISWOLD V. CONNECTICUT
381 U. S. 479 (1964)
H.L. V. MATHESON
450 U.S. 398. 425 (1981)
1900
JAN. 1921
JAN. 1958

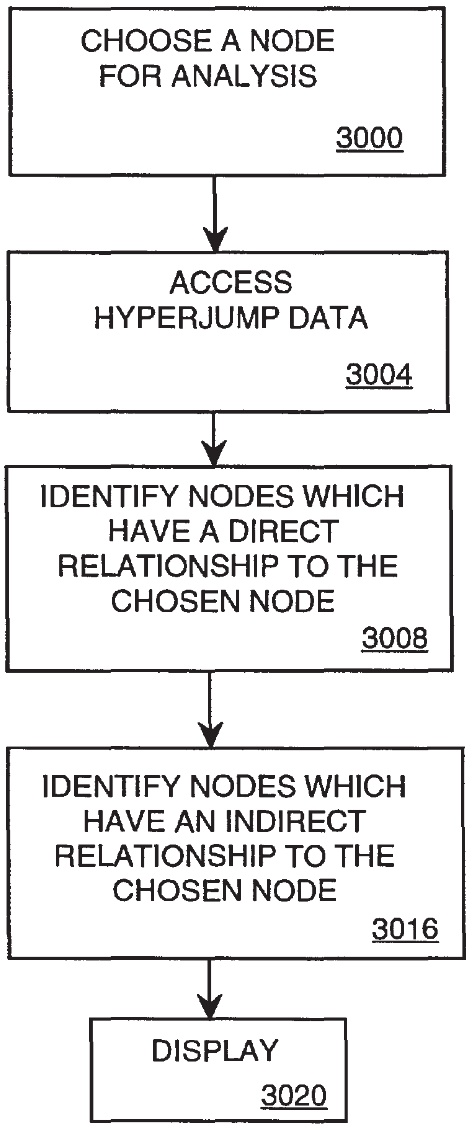


Fig. 14A

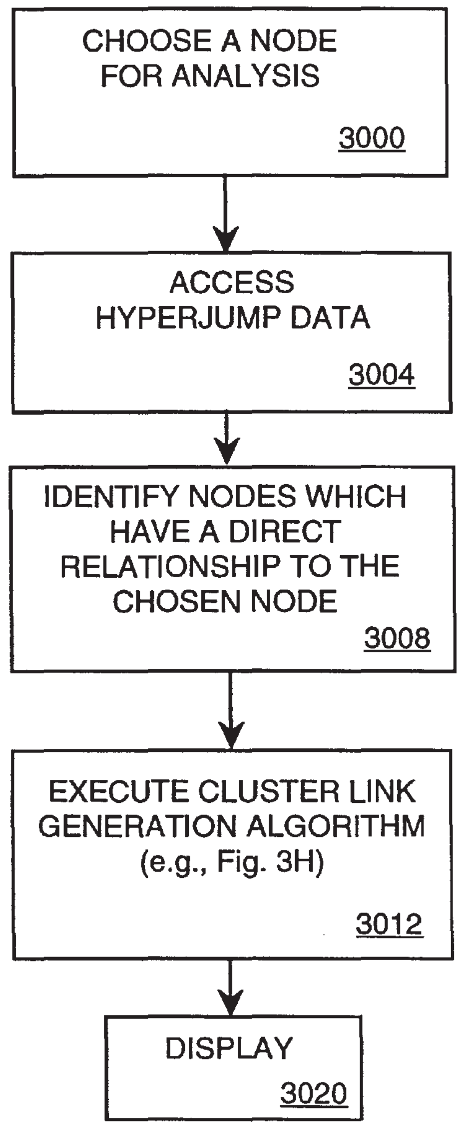


Fig. 14B

METHOD AND APPARATUS FOR INDEXING, SEARCHING AND DISPLAYING DATA

RELATED APPLICATIONS

This application is a divisional of U.S. patent application Ser. No. 08/649,304, filed May 17, 1996 entitled METHOD APPARATUS FOR INDEXING, SEARCHING AND DISPLAYING DATA, now U.S. Pat. No. 5,832,494, (which is hereby incorporated by reference) which is a continuation-in-part of U.S. application Ser. No. 08/076,658, filed Jun. 14, 1993 with the same title now U.S. Pat. No. 5,544,352.

TECHNICAL FIELD

This invention pertains to computerized research tools. More particularly, it relates to computerized research on databases. Specifically, the invention indexes data, searches data, and graphically displays search results with a user interface.

BACKGROUND

Two manuals containing background materials are hereby incorporated by reference "V-Search Integration Tool Kit For Folio VIEWS", containing thirty-six (36) pages, "V-Search Publisher's Tool Kit User's Manual", containing one hundred sixty (160) pages.

Our society is in the information age. Computers maintaining databases of information have become an everyday part of our lives. The ability to efficiently perform computer research has become increasingly more important. Recent efforts in the art of computer research have been aimed at reducing the time required to accomplish research. Computer research on non-textual objects is very limited. Current computer search programs use a text-by-text analysis procedure (Boolean Search) to scan a database and retrieve items from a database. The user must input a string of text, and the computer evaluates this string of text. Then the computer retrieves items from the database that match the string of text. The two popular systems for computerized searching of data used in the legal profession are Westlaw™, a service sold by West Publishing Company, 50 W. Kellogg Blvd., P.O. Box 64526, St. Paul, Minn. 55164-0526, and Lexis™, a service sold by Mead Data Central, P.O. Box 933, Dayton, Ohio 45401.

However, Boolean searches of textual material are not very efficient. Boolean searches only retrieve exactly what the computer interprets the attorney to have requested. If the attorney does not phrase his or her request in the exact manner in which the database represents the textual object, the Boolean search will not retrieve the desired textual object. Therefore, the researcher may effectively by denied access to significant textual objects that may be crucial to the project on which the researcher is working. A second problem encountered with Boolean searches is that the search retrieves a significant amount of irrelevant textual objects. (It should be noted that in the context of research, a textual object could be any type of written material. The term textual object is used to stress the fact that the present invention applies to all types of databases. The only requirement that a textual object must satisfy in order to be selected by a Boolean search program is that part of the textual object match the particular request of the researcher. Since the researcher cannot possibly know all of the groupings of text within all the textual objects in the database, the researcher is unable to phrase his request to only retrieve the textual objects that are relevant.

Aside from the inefficiency of Boolean searches, the present systems for computerized searching of data are inadequate to serve the needs of a researcher for several other reasons. Even if one assumes that all the textual objects retrieved from a Boolean search are relevant, the listing of the textual objects as done by any currently available systems does not convey some important and necessary information to the researcher. The researcher does not know which textual objects are the most significant (i.e., which textual object is referred to the most by another textual object) or which textual objects are considered essential precedent (i.e., which textual objects describe an important doctrine).

In the legal research field, both Westlaw™ and Lexis™ have a Shepardizing™ feature that enables the researcher to view a list of textual objects that mention a particular textual object. The Shepardizing feature does not indicate how many times a listed textual object mentions the particular textual object. Although the Shepardizing feature uses letter codes to indicate the importance of a listed textual object (e.g., an "f" beside a listed textual object indicates that the legal rule contained in particular textual object was followed in the listed textual object), data on whether a listed textual object followed the rule of a particular textual object is entered manually by employees of Shepard's™/McGraw Hill, Inc., Div. of McGraw-Hill Book Co., 420 N. Cascade Ave., Colorado Springs, Colo. 80901, toll free 1-800-525-2474. Such a process is subjective and is prone to error.

Another legal research system that is available is the Westlaw™ key number system. The Westlaw™ key number system has problems similar to the shepardizing feature on the Lexis™ and Westlaw™ systems.

The video displays of both the West™ and Lexis™ systems are difficult to use. The simple text displays of these systems do not provide a researcher with all the information that is available in the database.

Computerized research tools for legal opinions and related documents are probably the most sophisticated computer research tools available and therefore form the background for this invention. However, the same or similar computer research tools are used in many other areas. For example, computer research tools are used for locating prior art for a patent application. The same problems of inefficiency discussed above exist for computer research tools in many areas of our society.

What is needed is a system for computerized searching of data that is faster than the available systems of research.

What is needed is a system for computerized searching of data that enables researchers to research in a manner in which they are familiar.

What is needed is a computerized research tool that will reorganize, re-index or reformat the data into a more efficient format for searching.

What is needed are more sophisticated methods to search data.

What is needed is a system for computerized searching of data that will significantly reduce the number of irrelevant textual objects it retrieves.

What is needed is a user friendly computerized research tool.

What is needed is a visual user interface which can convey information to a user conveniently.

What is needed is a system for computerized searching of data that easily enables the researcher to classify the object according to his or her own judgment.

What is needed is a system for computerized searching of data that provides a visual representation of "lead" objects and "lines" of objects, permitting a broad overview of the shape of the relevant "landscape."

What is needed is a system for computerized searching of data that provides an easily-grasped picture or map of vast amounts of discrete information, permitting researchers to "zero in" on the most relevant material.

What is needed is a system for computer searching of data that provides a high degree of virtual orientation and tracking, the vital sense of where one has been and where one is going, and that prevent researchers from becoming confused while assimilating a large amount of research materials.

Accordingly, there is an unanswered need for a user friendly computerized research tool. There is a need for "intelligent" research technology that emulates human methods of research. There is a need in the marketplace for a more efficient and intelligent computerized research tool.

The present invention is designed to address these needs.

SUMMARY OF THE INVENTION

This invention is a system for computerized searching of data. Specifically, the present invention significantly aids a researcher in performing computerized research on a database or a network. The invention simplifies the research task by improving upon methods of searching for data including textual objects, and by implementing a user interface that significantly enhances the presentation of the data.

The invention can be used with an existing database by indexing the data and creating a numerical representation of the data. This indexing technique called proximity indexing generates a quick-reference of the relations, patterns, and similarity found among the data in the database. Using this proximity index, an efficient search for pools of data having a particular relation, pattern or characteristic can be effectuated. This relationship can then be graphically displayed.

There are three main components to the invention: a data indexing applications program, a Computer Search Program for Data Represented by Matrices ("CSPDM"), and a user interface. Each component may be used individually. Various indexing application programs, CSPDMs, and user interface programs can be used in combination to achieve the desired results. The data indexing program indexes data into a more useful format. The CSPDM provides efficient computer search methods. The preferred CSPDM includes multiple search subroutines. The user interface provides a user friendly method of interacting with the indexing and CSPDM programs. The preferred user interface program allows for easy entry of commands and visual display of data via a graphical user interface.

The method which the invention uses to index textual objects in a database is called Proximity Indexing. This method can also be used to index objects located on a network. The application of this method to network domains is discussed in greater detail later in this specification. Proximity Indexing is a method of preparing data in a database for subsequent searching by advanced data searching programs. Proximity Indexing indexes the data by using statistical techniques and empirically developed algorithms. The resulting search by an advanced data searching program of the Proximity Indexed data is significantly more efficient and accurate than a simple Boolean search.

The Proximity Indexing Application Program indexes (or represents) the database in a more useful format to enable the Computer Search Program for Data Represented by Matrices (CSPDM) to efficiently search the database. The Proximity Indexing Application Program may include one or more of the following subroutines: an Extractor: a Patterner: and a Weaver. The Proximity Indexing Application Program indexes (or represents) data in a locally located database or remotely located database. The database can contain any type of data, including text, alphanumerics, or graphical information.

In one embodiment, the database is located remotely from the Computer Processor and contains some data in the form of textual objects. The Proximity Indexing Application Program indexes the textual objects by determining how each full textual object (e.g., whole judicial opinion, statute, etc.) relates to every other full textual object by using empirical data and statistical techniques. Once each full textual object is related to each other full textual object, the Proximity Indexing Application Program compares each paragraph of each full textual object with every other full textual object as described above. The Proximity Indexing Application Program then clusters related contiguous paragraphs into sections. Subsequently, the Proximity Indexing Application Program indexes each section and the CSPDM evaluates the indexed sections to determine which sections to retrieve from the database. Such organization and classification of all of the textual objects in the database before any given search commences significantly limits the number of irrelevant textual objects that the CSPDM program retrieves during the subsequent search and allows retrieval of material based on its degree of relevancy.

In a preferred embodiment, the Proximity Indexing Application Program includes a link generation subroutine wherein direct and indirect relationships between or among data is used to generate a representation of the data. Generally, direct and indirect relationships in the database are identified as links and placed in a table.

Again, this method of computerized research can be used for nearly any database including those containing non-textual material, graphical material, newspapers material, data on personal identification, data concerning police records, etc.

The remaining two programs in the present invention are the CSPDM and the GUI Program. The CSPDM has seven subroutines that each search for different pools of objects. The GUI Program also has seven subroutines. Each CSPDM subroutine performs a different type of search. Each of the subroutines of the GUI uses the results of the corresponding subroutine of the CSPDM to create the proper display on the display.

After the Proximity Indexing Application Program indexes a database, the CSPDM application program is used to search the indexed database. For example, the CSPDM program can either be located in memory that is remote from the Computer Processor or local to the Computer Processor. In addition, the CSPDM program can either be remote or local in relation to the database.

The subroutines of the CSPDM utilize the coefficients and other data created by the Proximity Indexing Application Program to facilitate its search. However, if the researcher does not have the particular object citation available, the researcher can perform a Boolean search to retrieve and organize a pool of objects. Alternatively, the researcher can subsequently search for related objects by using the Pool-Similarity Subroutine, the Pool-Paradigm Subroutine, the Pool-Importance Subroutine or the Pool-Paradigm-Similarity Subroutine as defined below.

If the researcher already has the citation of a particular object available, the researcher can search for related objects by utilizing the Cases-In Subroutine, Cases-After Subroutine or Similar-Cases Subroutine. The Cases-In Subroutine retrieves all of the objects from the database to which a selected object refers. In addition, the subroutine determines the number of times the selected object refers to each retrieved object and other characteristics of each object, including its importance, and degree of relatedness to the selected object.

The Cases-After Subroutine retrieves all of the objects from the database that refer to the selected object. Also, the subroutine determines the number of times each retrieved object refers to the selected object and other characteristics of each object, including its importance and degree of relatedness to the particular object to which it refers.

The Similar-Cases Subroutine determines the degree of similarity between the retrieved objects and the selected object. Similarity may be defined, in the context of legal cases, as the extent to which the two objects lie in the same lines of precedent or discuss the same legal topic or concept. Numerous other relationships may be used to define similarity.

In addition, for a textual, object, if the researcher does not know of a particular textual object on which to base his or her search, the researcher may execute a Boolean word search. After a standard Boolean word search has been run, the researcher may run the Pool-Similarity Subroutine to retrieve information containing the degree of similarity between each textual object in the pool and a particular textual object selected by the user. Similarly, the Pool-Importance Subroutine can be used to determine the degree of importance (i.e., whether a judicial opinion is a Supreme Court opinion or a District Court opinion) and other characteristics of each textual object retrieved using the Boolean word search.

The Pool-Paradigm Subroutine calculates the geographic center in vector space of the pool of textual objects retrieved by the Boolean word search or other pool generating method. It then orders the retrieved textual objects by their degree of similarity to that center or "paradigm." The researcher can then evaluate this "typical textual object" and utilize it to help him or her find other relevant textual objects. In addition, the researcher can scan through neighboring "typical textual objects" to evaluate legal subjects that are closely related to the subject of the researcher's search.

The Pool-Paradigm-Similarity Subroutine similarly creates a paradigm textual object from the retrieved textual objects. However, the subroutine calculates the similarity of all textual objects in the database to the paradigm textual object in addition to the similarity of the retrieved textual objects to the paradigm textual object.

After the CSPDM has retrieved the desired objects, the Graphical User Interface (GUI) Program may be used to display the results of the search on the display. In one embodiment, the GUI is a user interface program. The GUI Program contains three main subroutines: Cases-In Display Subroutine (CIDS), Cases-After Display Subroutine (CADS) and Similar-Cases Display Subroutine (SCDS). The main subroutines receive information from the corresponding subroutines Cases-In, Cases-After and Similar-Cases of the CSPDM. The GUI Program also contains four secondary subroutines: Pool-Similarity Display Subroutine ("PSDS"), Pool-Paradigm Display Subroutine ("PPDS"), Pool-Importance Display Subroutine ("PIDS"), and the Pool-Paradigm-Similarity Subroutine (PPSDS). The secondary subroutines also receive information from the corresponding subroutines Pool-Similarity Subroutine, Pool-Paradigm Subroutine, Pool-Importance Subroutine and the Pool-Paradigm Similarity Subroutine of the CSPDM.

The CIDS subroutine receives information gathered from the Cases-In Subroutine of the CSPDM. The CIDS subroutine displays user friendly active boxes and windows on the display which represent the textual objects retrieved from the database represented in Euclidean space. It can also use the boxes to represent objects retrieved from a network. Various active box formats and arranging of information within the boxes may be utilized. The display depicts the appropriate location of textual objects in Euclidean space on a coordinate means. An algorithm may be used to determine the appropriate location of the boxes. The coordinate means may have one or more axis. In one embodiment, the horizontal axis of the coordinate means may represent the time of textual object creation; the vertical axis could represent a weighted combination of the number of sections in which that particular retrieved text is cited or discussed, its degree of importance, and its degree of similarity to the host textual object and the depth axis (Z-axis) represents the existence of data and length of the textual data or object.

The invention can also alter the background color of the window itself to communicate additional information graphically to the user. For example, if the horizontal axis represented time, then the invention could display the portion of the window containing objects occurring previous to the search object in one color and the portion containing the objects occurring after in another. Thus, the researcher can understand at a glance the relative position of his search target in relation to all the other objects related to it.

CIDS also enables the researcher to open up various active boxes on the display by entering a command into the computer processor with the input means. After entering the proper command, the active box transforms into a window displaying additional information about the selected textual object. These windows can be moved about the display and stacked on top or placed beside each other via the input means to facilitate viewing of multiple windows of information simultaneously. In one embodiment, the windows are automatically arranged by the computer system. Since the number of textual objects retrieved in a single search may exceed the amount which could be displayed simultaneously, the GUI Program enables the researcher to "zoom in" or "zoom out" to different scales of measurement on both the horizontal and vertical axis.

The CADS receives information gathered by the Cases-After Subroutine of the CSPDM. The CADS creates a display similar to the CIDS display. However, the active boxes representing the retrieved textual objects indicate which textual objects in the database refer to a selected textual object as opposed to which textual objects a selected textual object refers.

The SCDS receives information gathered by the Similar-Cases Subroutine of the CSPDM. The SCDS causes a similar display on the display as the CIDS and the CADS except that the vertical axis indicates the degree of similarity between the retrieved textual objects and the selected textual object.

The GUI Program contains four secondary subroutines: Pool-Search Display Subroutine (PSDS), Pool-Paradigm Display Subroutine (PPDS), Pool-Importance Display Subroutine (PIDS) and the Pool-Paradigm-Similarity Display Subroutine (PPSDS). The PSDS receives the results gath-

ered by the Pool-Search Subroutine of the CSPDM. The PPDS receives the results gathered by the Pool-Paradigm Subroutine of the CSPDM. The PIDS receives the results gathered by the Pool-Importance Subroutine of the CSPDM. The PPSDS receives the results gathered by the Pool-Paradigm-Similarity Subroutine of the CSPDM. The results of the PSDS, PPDS, PIDS and PPSDS are then displayed in a user friendly graphical manner similar to the results of the CIDS, CADS and SCDS. A researcher can access the PSDS, PIDS, PSDS or PPSDS from any of the three main or four secondary subroutines of the GUI to gather information corresponding to the active boxes that represent the pool of textual objects retrieved by the corresponding subroutine of the CSPDM.

By using the graphical display, the researcher can view immediately a visual representation of trends in the data (for example, trends developing in the law and current and past legal doctrines). In addition, the researcher can immediately identify important data or important precedent and which object serving as the precedent is most important to the project on which the researcher is working. This visual representation is a vast improvement over the current computerized research tools. Furthermore, the researcher using the present invention does not have to rely on the interpretation of another person to categorize different textual objects because the researcher can immediately visualize the legal trends and categories of law. In addition, new topic areas can be recognized without direct human intervention. The current research programs require a researcher to view objects in a database or to read through the actual text of a number of objects in order to determine which objects are important, interrelated, or most closely related to the topic at hand and which ones are not.

It is an object of this invention to create an efficient and intelligent system for computerized searching of data that is faster than available systems of research.

It is an object of the invention to integrate the system of computerized searching into the techniques to which researchers are already accustomed.

It is an object of the invention to utilize statistical techniques along with empirically generated algorithms to reorganize, re-index and reformat data in a database into a more efficient model for searching.

It is an object of the invention to utilize statistical techniques along with empirically generated methods to increase the efficiency of a computerized research tool.

It is an object of the invention to create a system of computerized searching of data that significantly reduces the number of irrelevant objects retrieved.

It is an object of this invention to create a user friendly interface for computer search tools which can convey a significant amount of information quickly.

It is an object of the invention to enable the researcher to easily and immediately classify retrieved database objects according to the researcher's own judgment.

It is an object of the invention to provide a visual representation of "lead" objects and "lines" of objects, permitting a broad overview of the shape of the relevant "landscape."

It is an object of the invention to provide an easily-grasped picture or map of vast amounts of discrete information, permitting researchers to "zero in" on the most relevant material.

It is an object of the invention to provide a high degree of virtual orientation and tracking that enables a researcher to keep track of exactly what information the researcher has already researched and what information the researcher needs to research.

These and other objects and advantages of the invention will become obvious to those skilled in the art upon review of the description of a preferred embodiment, and the appended drawings and claims.

DESCRIPTION OF THE DRAWINGS

FIG. **1** is a high level diagram of the hardware for the system for computerized searching of data.

FIG. **2** is high level diagram of the software for the system for computerized searching of data. The three main programs are the Proximity Indexing Application Program, the Computer Search Program for Data Represented by Matrices (CSPDM) Application Program and the Graphical User Interface (GUI) Program.

FIG. **3**A is a flow chart illustrating a possible sequence of procedures that are executed during the Proximity Indexing Application Program.

FIG. **3**B is a flow chart illustrating a possible sequence of the specific subroutines that are executed during one stage of the Proximity Indexing Application Program. The subroutines are the Initial Extractor Subroutine, Opinion Patterner Subroutine, the Opinion Weaver Subroutine, the Paragraph Patterner Subroutine (Optional), the Paragraph Weaver Subroutine and the Section Comparison Subroutine.

FIG. **3**C is flow chart illustrating a possible sequence of subroutines that are executed after the Section Comparison Subroutine. The Section Comparison Subroutine may comprise the Sectioner-Geographic Subroutine and the Section-Topical Subroutine (Optional). The sequence of subroutines executed after the Section Comparison Subroutine are the Section Extractor Subroutine, the Section Patterner Subroutine and the Section Weaver Subroutine.

FIG. **3**D is a high level flow chart illustrating a possible sequence of subroutines that comprise the Boolean Indexing Subroutine which are executed during another stage of the Proximity Indexing Application Program. The first two subroutines, Initialize Core English Words and Create p×w Boolean Matrix, are executed by the Initial Extractor Subroutine. The results are then run through the Pool-Patterner Subroutine, the Pool-Weaver Subroutine, the Pool-Sectioner Subroutine, the Section-Extractor Subroutine, the Section-Patterner Subroutine and the Section Weaver Subroutine.

FIG. **3**E is a chart illustrating the database format. The figure shows the types of structures contained within the database, links, link types, link subtypes, nodes, node types, node subtypes, and visual styles and also shows the various types of information that can be assigned to the links and nodes, including weights, identifications, names, comments, icons, and attributes.

FIG. **3**F is a high level diagram showing a sequence of nodes, N_o–N_3, connected by direct links which have weights W_1–W_3.

FIG. **3**G is a high level diagram showing a sequence of nodes, N_1–N_3, connected by direct and indirect links. The set of cluster links are also shown in the figure as functions of the weights associated with the direct links and the weight of the previous cluster link.

FIG. **3**H is a flow chart which depicts the Cluster Link Generation Algorithm.

FIG. **4**A is a high level diagram illustrating the flow of various search routines depending on the type of search initiated by the user by inputing commands to the Computer

Processor via the input means. The diagram further illustrates the interaction between the CSPDM and the GUI Program.

FIG. **4**B is a high level flow chart illustrating the sequence of subroutines in the CSPDM program and user interactions with the subroutines.

FIG. **4**C is a high level flow chart for the Cases-In Subroutine.

FIG. **4**D is a high level flow chart for the Cases-After Subroutine.

FIG. **4**E is a high level flow chart for the Similar-Cases Subroutine.

FIG. **4**F is a high level flow chart for the Pool-Similarity Subroutine.

FIG. **4**G is a high level flow chart for the Pool-Paradigm Subroutine.

FIG. **4**H is a high level flow chart for the Pool-Importance Subroutine.

FIG. **4**I is a high level flow chart showing two possible alternate Pool-Paradigm-Similarity Subroutines.

FIG. **5**A is a high level diagram illustrating the interaction between respective subroutines of the CSPDM and of the GUI Program. The diagram further illustrates the interaction between the GUI Program and the display.

FIG. **5**B is an example of the display once the Cases-After Display Subroutine (CADS) is executed.

FIG. **5**C is an example of the display after a user selects an active box representing a textual object retrieved by the Cases-After Subroutine and chooses to open the "full text" window relating to the icon.

FIG. **5**D is an example of the display once the Cases-In Display Subroutine (CIDS) is executed.

FIG. **5**E is an example of the display once the Similar-Cases Display Subroutine (SCDS) is executed.

FIG. **5**F is an example of the display after a user chooses to execute the Similar Cases Subroutine for a textual object retrieved by the Similar-Cases Subroutine represented in FIG. **5**E.

FIG. **5**G is an example of the display after a user chooses to execute the Similar Cases Subroutine for one of the cases retrieved by the Similar-Cases Subroutine represented in FIG. **5**F.

FIG. **5**H depicts an Executive Search Window.

FIG. **6** depicts a schematic representation of eighteen patterns.

FIG. **7** is a high level diagram of the Layout of Boxes Algorithm.

FIG. **8** is a diagram of a screen showing execution of a show usage command.

FIG. **9** is a diagram of the Internal Box Layout Algorithm.

FIG. **10**A is a diagram of a screen showing an Influence Map, which is a screen used in one embodiment of this invention.

FIG. **10**B is a diagram of a screen showing a Source Map, which is a screen used in one embodiment of this invention.

FIG. **10**C is a diagram of a screen showing a Cluster Map, which is a screen used in one embodiment of the invention.

FIG. **11** depicts a Look-Up Table for Bitmaps.

FIG. **12** is a software flow chart for the auto arranging window feature.

FIG. **13**A is a depiction of a display with vertically tiled windows.

FIG. **13**B is a depiction of a display with horizontally tiled windows.

FIG. **14**A is a high level diagram of a method for searching, indexing, and displaying data stored in a network.

FIG. **14**B is a high level diagram of a method for searching, indexing, and displaying data stored in a network using the cluster generation algorithm.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Referring now to the drawings, the preferred embodiment of the present invention will be described.

FIG. **1** is an overview of the preferred embodiment of the hardware system **26** for computerized searching of data. The hardware system **26** comprises a Computer Processor **30**, a database **54** for storing data, input means, display **38**, and RAM **34**.

The Computer Processor **30** can be a processor that is typically found in Macintosh computers, IBM computers, portable PCs, clones of such PC computers (e.g. Dell computers), any other type of PC, or a processor in a more advanced or more primitive computing device. Parallel processing techniques may also be utilized with this invention.

The database **54** is connected to the Computer Processor **30** and can be any device which will hold data. For example, the database **54** can consist of any type of magnetic or optical storing device for a computer. The database **54** can be located either remotely from the Computer Processor **30** or locally to the Computer Processor **30**. The preferred embodiment shows a database **54** located remotely from the Computer Processor **30** that communicates with the personal computer **28** via modem or leased line. In this manner, the database **54** is capable of supporting multiple remote Computer Processors **30**. The preferred connection **48** between the database **54** and the Computer Processor **30** is a network type connection over a leased line. It is obvious to one skilled in the art that the database **54** and the Computer Processor **30** may be electronically connected in a variety of ways. In the preferred embodiment the database **54** provides the large storage capacity necessary to maintain the many records of textual objects.

The input means is connected to the Computer Processor **30**. The user enters input commands into the Computer Processor **30** through the input means. The input means could consist of a keyboard **46**, a mouse **42**, or both working in tandem. Alternatively, the input means could comprise any device used to transfer information or commands from the user to the Computer Processor **30**.

The display **38** is connected to the Computer Processor **30** and operates to display information to the user. The display **38** could consist of a computer monitor, television, LCD, LED, or any other means to convey information to the user.

The Random Access Memory (RAM **34**) is also connected to the Computer Processor **30**. The software system **60** for computerized searching of data may reside in the RAM **34**, which can be accessed by the Computer Processor **30** to retrieve information from the software routines. A Read Only Memory (ROM), Erasable Programmable Read Only Memory (EPROM), disk drives, or any other magnetic storage device could be used in place of the RAM **34**. Furthermore, the RAM **34** may be located within the structure of the Computer Processor **30** or external to the structure.

The hardware system **26** for computerized searching of data shown in FIG. **1** supports any one, or any combination,

of the software programs contained in the software system **60** for computerized searching of data. The software system **60** for the computerized searching of data comprises one or more of the following programs: the Proximity Indexing Application Program **62**, the Computer Search Program for Data Represented by Matrices (CSPDM **66**) and the Graphical User Interface (GUI **70**) Program. The Proximity Indexing Application Program **62** could reside in RAM **34** or in separate memory **58** connected to the database **54**. The Computer Processor **30** or a separate computer processor **50** attached to the database **54** could execute the Proximity Indexing Application Program **62**. In the preferred embodiment the Proximity Indexing Application Program **62** resides in separate memory **58** that is accessible to the database **54**, and a separate computer processor **50** attached to the database **54** executes the Proximity Indexing Application Program **62**.

The CSPDM **66** could reside in the RAM **34** connected to the Computer Processor **30** or in the separate memory connected to the database **54**. In the preferred embodiment, the CSPDM **66** is located in the RAM **34** connected to the Computer Processor **30**. This is also the preferred embodiment for the application of this method to network searching. For network application, a separate database **54** storing information to be analyzed is remotely connected to the computer processor **30**. The CSPDM **66** may use the display **38** to depict input screens for user entry of information.

The GUI Program **70** could likewise reside in the RAM **34** connected to the Computer Processor **30** or in separate memory **58** connected to the database **54**. In the preferred embodiment, the GUI Program **70** is located in the RAM **34** connected to the Computer Processor **30**. The GUl Program **70** also communicates with the display **38** to enhance the manner in which the display **38** depicts information.

FIG. **2** is an overview of the preferred embodiment of the software system **60** for computerized searching of data. The software system **60** for computerized searching of data comprises at least one or more of the following programs: the Proximity Indexing Application Program **62**, the Computer Search Program for Data Represented by Matrices (CSPDM **66**) and the Graphical User Interface (GUI **70**) Program. Proximity Indexing is a method of identifying relevant data by using statistical techniques and empirically developed algorithms. (See Appendix #2) The Proximity Indexing Application Program **62** is an application program which represents or indexes the database **54** to a proper format to enable the Computer Search Program for Data Represented by Matrices (CSPDM **66**) to properly search the database **54**. The Proximity Indexing Application Program **62** can index data in a local database **54** or a remote database **54**. The Proximity Indexing Application Program **62** is shown in more detail in FIGS. **3**A to **3**H.

After the Proximity Indexing Application Program **62** indexes the database **54**, the CSPDM **66** application program can adequately search the database **54**. The CSPDM **66** program searches the database **54** for objects according to instructions that the user enters into the Computer Processor **30** via the input means. The CSPDM **66** then retrieves the requested objects. The CSPDM **66** either relays the objects and other information to the GUI Program **70** in order for the GUI Program **70** to display this information on the display **38**, or the CSPDM **66** sends display commands directly to the Computer Processor **30** for display of this information. However, in the preferred embodiment, the CSPDM **66** relays the objects and other commands to the GUI Program **70**. The CSPDM **66** is described in more detail in FIGS. **4**A to **4**I.

After the CSPDM **66** has retrieved the objects, the Graphical User Interface (GUI **70**) Program, which is a user interface program, causes the results of the search to be depicted on the display **38**. The GUI Program **70** enhances the display of the results of the search conducted by the CSPDM **66**. The GUI Program **70**, its method and operation, can be applied to other computer systems besides a system for computerized searching of data. The GUI Program **70** is described in more detail in FIGS. **5**A to **5**H.

FIGS. **3**A to **3**D depict examples of the procedures and subroutines of a Proximity Indexing Application Program **62**, and possible interactions among the subroutines. FIG. **3**A depicts a sequence of procedures followed by the Proximity Indexing Application Program **62** to index textual objects for searching by the CSPDM **66**. FIG. **3**B depicts specific subroutines that the Proximity Indexing Application Program **62** executes to partition full textual objects into smaller sections. FIG. **3**C depicts subroutines executed by the Section Comparison Routine of FIG. **3**B and subsequent possible subroutines to format and index the sections. FIG. **3**D depicts a sequence of subroutines of the Proximity Indexing Application Program **62** which first sections and then indexes these sections of "core english words" **140** contained in the database **54**. "Core english words" **140** are words that are uncommon enough to somewhat distinguish one textual object from another. The word searches of the CSPDM **66** search these sections of core English words to determine which textual objects to retrieve.

FIGS. **3**E–**3**H show a preferred embodiment for representing the data in a database **54** or documents in a network in accordance with the present invention. The application of this method for representing documents on a network is described in greater detail later in this specification.

FIG. **3**E shows a method for representing the data using the present invention. Specifically, FIG. **3**E shows a method in which links **2004** and nodes **2008** can be used along with link types **2012**, link subtypes **2020**, node types **2016** and node subtypes **2024** to represent the data.

A node **2008** is any entity that can be represented by a box on a display **38** such as a GUI **70**. A node **2008** might be, for example, an object in a database **54**, a portion of an object in a database **54**, a document, a section of a document, a World Wide Web page, or an idea or concept, such as a topic name. A node **2008** need not represent any physical entity such as an actual document. It is preferred that a node **2008** have links **2004**, specifically, it is preferred that a node **2008** have links to other nodes **2008** (for example source links (a source link is a link **2004**), or influence links (an influence link is a link **2004**)). A node **2008** can represent any idea or concept that has links to other ideas or concepts. For example, two nodes **2008** can exist such as a node **2008** called Modem Architecture (not shown) and a node **2008** called Classical Architecture (not shown) and the links would show that Classical Architecture is a source for Modern Architecture and that Modem Architecture is influenced by Classical Architecture. In this example, a source link **2004** and an influence link **2004** would exist between the two nodes **2008**. (Many times, links **2004** represent inverse relationships such as source links **2004** and influence links **2004**, and one type of link may be derived or generated from analysis of another link.)

More specifically, in the preferred embodiment, the software defines a node **2010** as something that has a unique node **2008** identification, a node type **2016**, a node subtype **2024**, and an associated date (or plot date **2011**). Node types **2016** or subtypes **2024** may have names **2021** or

identifications, title descriptors **2026** and external attributes **2018**. A node **2008** may have a corresponding numerical representation assigned, a vector, a matrix, or a table. In the preferred embodiment a table format is used for the nodes.

Referring to FIGS. **3**E, **3**F, and **3**G, a link **2004** is another name or identification for a relationship between two nodes **2008**. The relationship may be semantical, non-semantical, stated, implied, direct **2032**, indirect **2036**, actual, statistical and/or theoretical. A link **2004** can be represented by a vector or an entry on a table and contain information for example, a from-node identification **2010** (ID), a to-node ID **2010**, a link type **2012**, and a weight **2034**. A group of links **2004** may be represented by a series of vectors or entries in a table, a link table. Link subtypes **2020** may be used, named and assigned comments.

In addition, to better integrate the GUI **70** and the data representation, visual styles **2028** may be assigned for example to nodes **2008**, links **2004**, link types **2012**, and link subtypes **2020** to assist in the visual displays **38**.

In the preferred embodiment, three types of links **2004** are used: source links **2004**, influence links **2004** and cluster links **2004**. Source links **2004** generally link a first node **2008** to second node **2008** that represents information or documentation specifically cited or referred to by the first node **2008**. Influence links **2004** are generally the inverse of a source link **2004**. The relationships represented by these links **2004** may be explicit or implied.

Links **2004** and nodes **2008** may be manually entered by a user or automatically generated by a computer **30**. It is preferred that duster links **2004** be generated automatically by a processor. A cluster link **2004** is a relationship between two nodes **2008**, for example, two nodes **2008** both directly linked to the same intermediate nodes **2008**, may be indirectly linked through many paths and therefore have a cluster link **2004** between them. The cluster links **2004** may be determined using the specific or general methods described later for finding relationships in a database **54**. However, the preferred method is through using a Proximity Indexing Application Program **62**.

"Proximity indexing" is a method of indexing that uses statistical techniques and empirically generated algorithms to organize and categorize data stored in databases or on a network. The Proximity Indexing Application Program **62** applies the Proximity indexing method to a database **54**. One embodiment of the present invention uses the Proximity Indexing Application Program **62** to Proximity index textual objects used for legal research by indexing objects based on their degree of relatedness—in terms of precedent and topic—to one another.

Applying the method to legal research, the "Proximity indexing" system treats any discrete text as a "textual object." Textual objects may contain "citations," which are explicit references to other textual objects. Any legal textual object may have a number of different designations of labels. For example, 392 U.S. 1, 102 S.Ct 415, 58 U.S.L.W. 1103, etc. may all refer to the same textual object.

Cases are full textual objects that are not subsets of other textual objects. Subjects of a full textual object include words, phrases, paragraphs, or portions of other full textual objects that are referred to in a certain full textual object. (The system does not treat textual objects as subsets of themselves.)

Every case, or "full" textual object, is assigned a counting-number "name" —designated by a letter of the alphabet in this description—corresponding to its chronological order in the database **54**. Obviously, textual objects may contain citations only to textual objects that precede them. In other words, for full textual objects, if "B cites A," (i.e. "A is an element of B" or "the set 'B' contains the name 'A'"), textual object A came before B, or symbolically, A<B. Every textual object B contains a quantity of citations to full textual objects, expressed as Q(B), greater than or equal to zero, such that Q(B)<B.

Textual objects other than full textual objects may be subsets of full textual objects and of each other. For example, a section, page, or paragraph of text taken from a longer text may be treated as a textual object. Phrases and words are treated as a special kind of textual object, where Q(w)=0. Sections, pages, and paragraphs are generally subsets of only one full textual object, and may be organized chronologically under the numerical "name" of that full textual object. For purposes of chronology, phrases and words are treated as textual objects that precede every full textual object, and can generally be treated as members of a set with name "0," or be assigned arbitrary negative numbers.

Any two textual objects may be related to each other through a myriad of "patterns." Empirical research demonstrates that eighteen patterns capture most of the useful relational information in a cross-referenced database **54**. A list of these eighteen patterns, in order of importance, follows:

Given that:

a, b, c<A;

A<d, e, f<B; and

B<g, h, i.

Patterns Between A and B Include

1. B cites A.
2. A cites c, and B cites c.
3. g cites A, and g cites B.
4. B cites f, and f cites A.
5. B cites f, f cites e, and e cites A.
6. B cites f, f cites e, e cites d, and d cites A.
7. g cites A, h cites B. g cites a, and h cites a.
8. i cites B, i cites f [or g], and f [or g] cites A.
9. i cites g, i cites A, and g cites B.
10. i cites g [or d], i cites h, g [or d] cites A, and h cites B.
11. i cites a, i cites B, and A cites a.
12. i cites A, i cites e, B cites e.
13. g cites A, g cites a, A cites a, h cites B, and h cites a.
14. A cites a, B cites d, i cites a, and i cites d.
15. i cites B, i cites d, A cites a, and d cites a.
16. A cites b, B cites d [or c], and d [or c] cites b.
17. A cites b, B cites d, b cites a, and d cites a.
18. A cites a, B cites b, d [or c] cites a, and d [or c] cites b.

These 18 patterns are shown schematically in FIG. **6**. (For a discussion on probability theory and statistics, see Wilkinson, Leland; SYSTAT: The System for Statistics; Evanston, Ill.: SYSTAT Inc., 1989 incorporated herein by reference.) Some patterns occur only between two full textual objects, and others between any two textual objects; this distinction is explained below.

Semantical patterning is only run on patterns number one and number two, shown above.

For purposes of explaining how patterns are used to generate the Proximity Index, only the two simplest patterns are illustrated.

The simplest, Pattern #1, is "B cites A." See FIG. **6**. In the notation developed, this can be diagramed: a b c A d e f B g h i, where the letters designate textual objects in chronological order, the most recent being on the right, arrows

above the text designate citations to A or B, and -arrows below the text designate all other citations. The next simplest pattern between A and B, Pattern #2, is "A cites c, and B cites c," which can also be expressed as "there exists c, such that c is an element of (A intersect B)." See Appendix #1. This can be diagramed: a b c A d e f B g h i. For every textual object c from 0 to (A−1), the existence of Pattern #2 on A and B is signified by 1, its absence by 0. This function is represented as P#2AB(c)=1 or P#2AB(c)=0. The complete results of P#1AB and P#2AB can be represented by an (A)×(1) citation vector designated X.

The functions of some Patterns require an (n)×(1) matrix, a pattern vector. Therefore it is simplest to conceive of every Pattern function generating an (n)×(1) vector for every ordered pair of full textual objects in the database **54**, with "missing" arrays filled in by 0s. Pattern Vectors can be created for Pattern #1 through Pattern #4 by just using the relationships among textual object A and the other textual objects in the database **54** and among textual object B and the other textual objects in the database **54**. Pattern Vectors for Patterns # 5 through # 18 can only be created if the relationship of every textual object to every other textual object is known. In other words, Pattern Vectors for Patterns # 1 through # 4, can be created from only the rows A and B to the Citation Matrix but Pattern Vectors for Patterns #5 through #18 can only be created from the whole Citation Matrix.

(total textual objects c)/(theoretical maximum textual objects c) $[(x)(x)^T/\text{TMax}]$,

(total textual objects c)/(actual maximum textual objects c) $[(x)(x)^T/\text{AMax}]$

frequency of object c per year [f], and

the derivative of the frequency [f'].

In pattern # 2, given that A<B, the theoretical maximum ("TMax") number Q(A intersect B)=A minus 1. The actual maximum possible ("AMax"), given A and B, is the lesser of Q(A) and Q(B). The ratios "$X(X)^T$/TMax" and "$X(X)^T$/AMax," as well as the frequency of occurrence of textual objects c per year, f**2**(A, B), and the first derivative f'**2**(A, B), which gives the instantaneous rate of change in the frequency of "hits," are all defined as "numerical factors" generated from patterns #1 and #2. These are the raw numbers that are used in the weighing algorithm.

For Pattern #2, the total number of possible textual objects c subject to analysis, i.e., TMax, is A−1, one only for the years at issue which are those up to the year in which A occurred. However, a relationship may remain "open," that is, it may require recalculation of f(x) and f'(x) as each new textual object is added to the database **54**, (for a total of n cases subject to analysis).

The "numerical factors" for all eighteen patterns are assigned various weights in a weighing algorithm used to generate a scalar F(A, B). The function F generates a scalar derived from a weighted combination of the factors from all eighteen patterns. The patterns are of course also weighted by "importance," allowing Supreme Court full textual objects to impose more influence on the final scalar than District Court full textual objects, for example. The weighing of the more than 100 factors is determined by empirical research to give results closest to what an expert human researcher would achieve. The weighing will vary depending upon the type of material that is being compared and the type of data in the database **54**. (See Thurstone. The Vectors of Mind, Chicago, Ill.: University of Chicago Press, 1935, for a description of factor loading and manipulating empirical data incorporated herein by reference.) In a commercial "Proximity Indexer" it will be possible to reset the algorithm to suit various types of databases.

A scalar F(A, B) is generated for every ordered pair of full cases in the database **54**, from F(1, 2) to F(n−1, n). F(z,z) is defined as equal to 0.

The full results of F(A,B) are arranged in an (n)×(n) matrix designated F. Note that F(B, A) is defined as equal to F(A, B), and arrays that remain empty are designated by 0. For every possible pairing of cases (A,B), a Euclidean distance D(A,B) is calculated by subtracting the Bth row of Matrix F from the Ath row of Matrix F. In other words:

$$D(A,B)=[(F(1,A)-F(1,B))^2+(F(2,A)-F(2,B))^2+\ldots+(F(n,A)-F(n,B))^2]^{1/2}.$$

A function designated D(A,B) generates a scalar for every ordered pair (A,B), and hence for every ordered pair of textual objects (A,B) in the database **54**. The calculations D(A,B) for every ordered pair from D(1,1) to D(n,n) are then arranged in an (n)×(n) "proximity matrix" D. Every column vector in D represents the relationship between a given case A and every other case in the database **54**. Comparing the column vectors from column A (representing textual object A) and column B (representing textual object B) allows one to identify their comparative positions in n-dimensional vector space, and generate a coefficient of similarity, S(A,B), from 0–100%, which is more precise and sophisticated than F(A,B) or D(A,B) alone. A similarity subroutine can run directly on F(A,B). However, the real power of the Proximity Matrix D is that it allows one to identify "groups" or "clusters" of interrelated cases.

Through factor loading algorithms, the relationships represented by D for "n" cases can be re-represented in a vector space containing fewer than "n" orthogonal vectors. This knowledge can be reflected in S(A,B).

The Proximity Indexing Application Program **62** is an application program that applies the above techniques and algorithms to index and format data to be searched by the CSPDM **66**.

FIG. **3**A describes the overall procedure of the Proximity Indexing Application Program **62**. The first stage initializes the data **74** in the database **54**. The second stage determines the relationships between full textual objects **78**. The third stage determines the relationships between paragraphs of each textual object and each full textual object **80**. The fourth stage clusters related paragraphs using factor loading and empirical data and then groups the paragraphs into sections based on such data **84**. The fifth stage determines the relationships between the sections **88**. In the final stage, the sectioned textual objects are not further processed until commands are received from the CSPDM Routine **92**.

The following description of FIG. **3**B and FIG. **3**C elaborates on this general procedure by describing specific subroutines of a Proximity Indexing Application Program **62**. The following is a step by step description of the operation of the Proximity Indexing Application Program **62**.

Section A Initial Extractor Subroutine **96**

FIG. **3**B describes subroutines for the first portion of the preferred Proximity Indexing Application Program **62**. The first subroutine of the Proximity Indexing Applications Program is the Initial Extractor Subroutine **96**. The Initial Extractor Subroutine **96** performs three primary functions: Creation of the Opinion Citation Matrix, creation of the Paragraph Citation Matrix, and creation of Boolean Word Index.

The following steps are performed by the Initial extractor subroutine **96**.

1. Number all full textual objects chronologically with arabic numbers from 1 through n.

2. Number all paragraphs in all the full textual objects using arabic numbers from 1 through p.

3. Identify the page number upon which each paragraph numbered in step two above begins.

4. Create Opinion Citation Vectors (X). By comparing each full textual object in the data base to every other full textual object in the data base that occurred earlier in time.

5. Combine Opinion Citation Vectors to create the bottom left half portion of the n×n Opinion citation matrix.

6. Create a mirror image of the bottom left half portion of the Opinion citation matrix in the top right half portion of the same matrix, to complete the matrix. In this manner only $n^2/2$ comparisons need to be conducted. The other ½ of the comparisons are eliminated.

7. Create the p×n Paragraph Citation Vectors by comparing each paragraph to each full textual object that occurred at an earlier time. This will require (n/2)p searches.

8. Create a Paragraph Citation Matrix by combining Paragraph Citation Vectors to create the bottom left half portion of the matrix.

9. Complete the creation of the Paragraph Citation Matrix by copying a mirror image of the bottom left half portion of the matrix into the top right half portion of the matrix.

10. Initialize the Initial Extractor Subroutine **96** with a defined set of core English words **140**.

11. Assign identification numbers to the core English words **140**. In the preferred embodiment 50,000 English words are used and they are assigned for identification the numbers from −50,000 to −1.

12. Create a Boolean Index Matrix **144** with respect to the core English words by searching the database **54** for the particular word and assigning the paragraph number of each location of the particular word to each particular word. This procedure is described in greater detail in FIG. **3**D.

Section B Opinion Patterner Subroutine **100**

The Opinion Patterner Subroutine **100** performs three primary functions: Pattern analysis 6n matrices, calculation of the numerical factors and weighing the numerical factors to reach resultant numbers.

13. Process the Opinion Citation Matrix through each of the pattern algorithms described above and in FIG. **6** for each ordered pair of full textual objects to create opinion pattern vectors for each pattern and for each pair of full textual objects. The pattern algorithms determine relationships which exist between the ordered pair of textual objects. The first four pattern algorithms can be run utilizing just the Opinion Citation Vector for the two subject full textual objects. Each pattern algorithm produces a opinion pattern vector as a result. The fifth through eighteenth pattern algorithms require the whole Opinion Citation Matrix to be run through the Opinion Patterner Subroutine **100**.

14. Calculate total hits (citation) for each pattern algorithm. This can be done by taking the resultant opinion pattern vector ($\underline{OPV}$) and multiplying it by the transposed opinion pattern vector $(\underline{OPV})^T$ to obtain a scalar number representing the total hits.

15. Calculate the theoretical maximum number of hits. For example, in the second pattern, the theoretical maximum is all of the full textual objects that occur prior in time to case A (A−1).

16. Calculate the actual maximum number of hits. For example, in the second pattern, the actual maximum possible number of hits is the lesser of the number of citations in full textual object Q(A) or full textual object Q(B).

17. Calculate the total number of hits (citations) per year. This is labeled f(A,B).

18. Calculate the derivative of the total change in hits per year. This is the rate of change in total hits per year and is labeled f' (A,B).

19. Calculate the ratio of total hits divided by theoretical max $[((O\underline{FV})(O\underline{VP})^{t/TMAX})$.

20. Calculate the ratio of the total hits divided by the actual maximum $[(O\underline{PV})(O\underline{PV})^{t}{}_{/AMAX}]$.

21. Calculate a weighted number F(A,B) which represents the relationship between full textual object A and full textual object B. The weighted number is calculated using the four raw data numbers, two ratios and one derivative calculated above in steps 14 through 20 for each of the 18 patterns. The weighing algorithm uses empirical data or loading factors to calculate the resulting weighted number.

22. The Opinion Patterner Subroutine **100** sequence for the Opinion Citation Matrix is repeated n−1 times to compare each of the ordered pairs of full textual objects. Therefore, during the process, the program repeats steps 13 through 21, n−1 times.

23. Compile the Opinion Pattern Matrix by entering the appropriate resulting numbers from the weighing algorithm into the appropriate cell locations to form an n×n Opinion Pattern Matrix.

Section C The Opinion Weaver Subroutine **104**

The Opinion Weaver Subroutine **104** shown in FIG. **3**B, performs two primary tasks: calculation of the Opinion Proximity Matrix and calculation of the Opinion Similarity Matrix. The Opinion Proximity Matrix D is generated by calculating the Euclidean Distance between each row A and B of the Opinion Pattern Matrix (D(A,B)) for each cell DC(A,B). The Opinion Similarity Matrix is generated by calculating the similarity coefficient from 0 to 100 between each row A and B of the Opinion Proximity Matrix (S(A,B)) in each cell SC(A,B) in matrix S.

24. Calculate the n×n Opinion Proximity Matrix. To calculate D(A,B) the program takes the absolute Euclidian distance between column A and column B of the n×n Opinion Pattern Matrix. The formula for calculating such a distance is the square root of the sum of the squares of the distances between the columns in each dimension, or:

$$D(A,B)=[(F(1,A)-F(1,B))^2+(F(2,A)-F(2,B))^2+\ldots+(F(N,A)F(N,B))^2]^{1/2}$$

The Opinion Proximity Matrix created will be an n×n matrix. The smaller the numbers in the Opinion Proximity Matrix the closer the relationship between full textual object A and full textual object B.

25. Create n×n Opinion Similarity Matrix. To calculate the Opinion Similarity Matrix each scalar number in the Opinion Proximity Matrix is processed through a coefficient of similarity subroutine which assigns it a number between 0 and 100. By taking the coefficient of similarity, the program is able to eliminate full textual objects which have Euclidian distances that are great. (For example, a Euclidean distance that is very large and is run through the coefficient of similarity would result in a very low coefficient of similarity. Euclidean distances resulting in similarities below four are eliminated in the preferred embodiment).

Section D Paragraph Patterner Subroutine **108** (Optional)

26. Obtain the p×n Paragraph Citation Matrix calculated by the Initial Extractor Subroutine **96**.

27. Run each ordered pair of rows of the p×n Paragraph Citation Matrix for an individual full textual object i through the pattern algorithms number one and two and determine the resultant Paragraph Pattern Vector.

28. Calculate the various numerical factors (AMax, TMax, etc.) by evaluating the values in the Paragraph Pattern Vector.

29. Run the Paragraph Pattern Vector and the numerical factors through the weighing algorithm to determine the

appropriate value for each cell of the c_i×n Partial Paragraph Pattern Matrix where c_i is the number of paragraphs in full textual object i.

30. Repeat steps 27 through 29 for each full textual object i where i=1 to n, to create the p×n Paragraph Pattern Matrix.

Section E Paragraph Weaver Subroutine **112**

31. Calculate the Euclidean distance of each ordered pair of rows of either the p×n Paragraph Citation Matrix or the p×n Paragraph Pattern Matrix for a single full textual object i.

32. Place the resultant Euclidean distance values in the appropriate cell of the c_i×c_i Paragraph Proximity Matrix where c_i is the number of paragraphs in full textual object i, where 0<i<n+1.

33. Repeat steps 31 through 32 n times in order to calculate n different Paragraph Proximity Matrices (one for each full textual object i).

34. The Section Comparison Subroutine **116** clusters all p paragraphs in the database **54** into sections. Then the sections are compared and indexed in the database **54**. This procedure is described in greater detail in FIG. **3**C.

FIG. **3**C depicts possible subroutines that the Section Comparison Subroutine **116** comprises. The subroutines are the Sectioner Geographical Subroutine **120**, the Sectioner Topical Subroutine **124** (Optional), the Section Extractor Subroutine **128**, the Section Patterner Subroutine **132** and the Section Weaver Subroutine **136**. Section F Sectioner Geographical Subroutine **120**

35. For each full textual object i, the Sectioner Geographical Subroutine **120** uses the corresponding c_i×c_i Paragraph Proximity Matrix and a contiguity factor for each paragraph to determine which paragraphs may be clustered into sections. Sections are made up of continuous paragraphs that are combined based upon weighing their Euclidean distances and contiguity.

36. Repeat step 35 for all n full textual objects until all p paragraphs are grouped into q sections.

Section H Sectioner Topical Subroutine **124** (Optional)

37. The Sectioner Topical Subroutine **124** provides additional assistance to the Sectioner Geographical Subroutine **120** by considering the factor of topical references to determine the q sections.

38. For the total number of discrete references "z" to each full textual object in a particular full textual object, a z×z Citation Proximity Matrix is formed by comparing the Euclidean distances between each reference to a full textual object contained in each paragraph and calculating the topical weight given to each paragraph.

Section I Section Extractor Subroutine **128**

39. The Section Extractor Subroutine **128** numbers each section created by the Sectioner Geographical Subroutine **120** and Sectioner Topical Subroutine **124** Subroutines from 1 to q.

40. The Sectioner Extractor Subroutine **128** creates a q×q Section Citation Matrix by determining which sections refer to every other section.

Section J Section Patterner Subroutine **132** (shown in FIG. **3**C)

41. The Section Patterner Subroutine **132** then calculates 18 Section Pattern Vectors corresponding to each row of the q×q Section Citation Matrix using the 18 pattern algorithms.

42. From the Section Pattern Vectors, the numerical factors (AMax, TMax, etc.) are calculated.

43. The weighing algorithm evaluates the numerical factors and the Section Pattern Vectors and determines the values for each cell of the q×q Section Pattern Matrix.

Section K Section Weaver Subroutine **136**

44. The Section Weaver Subroutine **136** calculates the Euclidean distances between each row of the q×q Section Pattern Matrix and creates a q×q Section Proximity Matrix.

45. The Section Weaver Subroutine **136** then creates a q×q Section Similarity Matrix with coefficients 0 to 100 using the values of the Section Proximity Matrix and empirical data and factor loading.

Section L Semantical Clustering of a Boolean Index Routine **138**

FIG. **3**D depicts a possible Semantical Clustering of a Boolean Index Routine **138**. (See Hartigan, J. A. *Clustering Algorithms.* New York: John Wiley & Sons, Inc., 1975, for detailed description of clustering algorithms incorporated herein by reference.) The Semantical Clustering routine of a Boolean Index **138** indexes the textual objects according to the similarity of phrases and words contained within each textual object in a database **54**. The routine comprises seven possible subroutines: the Initial Extractor Subroutine **96**, the Pool Patterner Subroutine **152**, the Pool Weaver Subroutine **96** the Pool Sectioner Subroutine **160**, the Section Extractor Subroutine **128**, the Section Patterner Subroutine **132** and the Section Weaver Subroutine **136**, In fact, it is quite possible, using only semantical statistical techniques, to "Proximity-index" documents that do not refer to one another at all based on there Boolean indices.

Section M Initial Extractor Subroutine **96**

46. As described in steps 10 and 11, the Initial Extractor Subroutine **96** initializes a set of core English words **140** and assigns each word a number. The preferred embodiment uses 50,000 discrete core English words and assigns each discrete core English word a number from −50,000 to −1.

47. The Initial Extractor Subroutine **96** then converts the core English words into a p×w matrix. The number of columns (w) represents the number of discrete core English words in the database **54** and the number of rows (p) represents the number of paragraphs in the database **54**.

48. The Initial Extractor Subroutine **96** fills the p×w matrix by inserting a "1" in the matrix cell where a certain paragraph contains a certain word.

Section N Pool Patterner Subroutine **152**

49. The Pool Patterner Subroutine **152** creates two pattern algorithm vectors for only the first two patterns and determines values for the total number of hits, the theoretical maximum number of hits, the actual maximum number of hits, the total number of hits per year and the derivative of the total number of hits per year.

50. The weighing algorithm of the Pool Patterner Subroutine **152** uses empirical data and factor loading to determine values to enter into a p×w Paragraph/Word Pattern Matrix.

51. The Pool Weaver Subroutine **156** creates a p×w Paragraph/Word Pattern Matrix by filling the appropriate cell of the Matrix with the appropriate value calculated by the weighing algorithm.

52. The Pool Patterner Subroutine **152** creates a p×w Paragraph/Word Proximity Matrix taking the Euclidean distance between the rows of the Paragraph/Word Pattern Matrix.

Section O Pool Sectioner Subroutine **160**

53. The Pool Sectioner Subroutine **160** evaluates the Euclidean distances in the Paragraph/Word Proximity Matrix and the contiguity factor of each paragraph to cluster the paragraphs (p) into a group of (v) sections and create a v×w Preliminary Cluster Word Matrix.

Section P Section Extractor Subroutine **128**

54. The-Section Extractor Subroutine **128** numbers each section chronologically and creates a v×v Section Word Citation Matrix.

Section Q Section Patterner Subroutine **132**

55. The Section Patterner Subroutine **132** evaluates the v×v Section Word Citation Matrix to create two word pattern vectors for only the first two patterns algorithms (described above and shown in FIG. **6**) and determines numerical factors for the total number of hits, the theoretical maximum number of hits, the actual maximum number of hits, the total number of hits per year and the derivative of the total number of hits per year.

56. The Weighing algorithm uses empirical data and factor loading to weigh the numerical factors created from the word pattern vectors and uses the numerical factors and the word pattern vectors to determine values t o enter into a v×v Section Word Pattern Matrix.

Section R Section Weaver Subroutine **136**

57. The Section Weaver Subroutine **136** creates a v×v Section Word Proximity Matrix by taking the Euclidean distance between the rows of the Section Word Pattern Matrix and placing the appropriate Euclidean distance value in the appropriate cell of the Section Word Proximity Matrix.

58. The Section Weaver Subroutine **136** create a v×v Section Word Similarity Matrix by evaluating the Euclidean distances from the Section Word Proximity Matrix and empirical data, and calculating the similarity coefficient for each order ed pair of sections, and places the value in the appropriate cell of the Section Word Similarity Matrix.

59. The Pool Searches of the CSPDM **66** evaluate the Section Word Similarity Matrix as well as other matrices to determine whether or not to retrieve a full textual object.

The following describes a preferred cluster link generator **2044** which implements a specific type of patterer or clustering system for use alone or in conjunction with other proximity indexing subroutines, and prior to searching. The cluster link generator **2044** analyzes a set of numerical representations of a database **54** and generates a second set of numerical representations of the database **54**. This second set is stored in the RAM **34**. This second set of numerical data can represent indirect **2036**, direct **2032**, or a combination of both direct **2032** and indirect **2036** relationships in the database **54**. Preferably, the second set of numerical representations accounts for indirect **2036** relationships in the database **54**. It is preferred that the first and second set of numerical data be in a table format and that-the first set represent direct **2032** relationships or links and the second set represent cluster links **2004**.

Referring to FIG. **3**H, the cluster link generation algorithm **2044** analyzes links to generate a set of cluster links **2004**. More specifically, the cluster link generation algorithm **2044** generates a set of cluster links **2004** by analyzing direct **2032** and/or indirect relationships **2036** between nodes **2008** or between objects in a database **54** and generates a set of cluster links **2004**.

In the preferred embodiment, the cluster link generator **2044** analyzes direct links **2004** (for example source links **2004** and influence links **2004**). These direct links **2032** may be represented by a table or series of vectors. The cluster link generator **2044** then locates indirect relationships **2036** between nodes **2008** or objects in a database **54**. The indirect relationships **2036** are preferably made up of direct links **2032**. The indirect relationship **2036** paths are preferably made up of direct links **2004**. The cluster link generator **2040** then generates a set of cluster links **2004** based upon both the direct links **2032** and on the indirect relationships. The set of cluster links **2004** may be represented by a table or a series of vectors. Another embodiment of this invention uses candidate cluster links **2004** to provide a more efficient search. Candidate cluster links are the set of all possible cluster links **2004** between a search node **2008** and a target node **2004**. In this embodiment, only a subset of the candidate cluster links **2004**, the actual cluster links **2004**, which meet a certain criteria are used to locate nodes **2008** for display.

Consider a set of nodes **2008** $N_0 \ldots N_3$ connected by a sequence of direct links **2032** whose weights **2034** are given by $W_1 \ldots W_3$, as shown in FIGS. **3**F.

Node **2008** N_1 is reachable from N_0 through a path P_1 of length **1** (that is, $N_0 \rightarrow N_1$); node **2008** N_2 is reachable through a path P_2 of length **2** ($N_0 \rightarrow N_1 \rightarrow N_2$); and so on.

Each path P provides some evidence that the start node **2008** (N_0) and destination node **2008** (N_1, N_2, or N_3) are related to some extent. The strength of the indirect relationship **2036** depends on a length L of the path P and on the weights **2034** of the individual direct links **2032** along that path P.

In FIG. **3**G, the indirect relationship **2036** from N_0 to N_1, N_2, and N_3 are shown as arcs.

The weight $C_1 \ldots C_3$of each implied relationship, is a function of the weight **2034** from the path to the previous node **2008** and the weight **2034** of the last direct link **2032**.

The individual functions F**1** . . . F**3** describe how to combine the weights **2034** of the direct links **2004** to determine the weight c of an indirect link **2036**. Selecting appropriate functions is the key to making cluster link generation work well. A preferred definition of F_N is as follows:

$$C_N = F_N(C_{N-1}, W_N) = \min(C_{N-1}, D_N * W_N),$$

where D_N is a damping factor that decreases rapidly as N increases.

The cluster link algorithm **2044** determines the set of all paths P from a given start node **2008** N_0 that have a length less than or equal to a given length L. Each path is rated using the method described above. The paths are then grouped by destination node **2008**; the candidate cluster link **2004** $C(N_0, N_N)$ between N_0 and a given destination node **2008** N_N has a weight C_N equal to the sum of the weights **2034** of all paths P_N leading to N_N.

The set of all candidate cluster links **2004** is then sorted by weight **2034**. A subset of the candidate links **2004** is chosen as actual cluster links **2004**. The number of cluster links **2032** chosen may vary, depending on the number of direct links **2004** from N_0, and on the total number of candidate cluster links **2004** available to choose from.

Performance considerations and efficiency are more important with large databases than for small databases. For large databases, finding the set of all paths P from a given node **2008** N_0 that have a length less than or equal to a given length L may be impractical, since the number of unique paths may number in the tens of millions.

One embodiment of this invention uses candidate cluster links **2004** to provide a more efficient search. Candidate cluster links **2004** are the set of all possible cluster links **2004** between a start node (**2008**) and a destination node (**2008**).

Clearly, it is not necessary to examine millions of paths when the goal is to select the top or strongest duster links **2004** for each start node **2008** N_0 (for example, the top 20 to 25 cluster links **2004**). The great majority of paths have an insignificant effect on the final results. What is needed is an implementation of the cluster link algorithm **2044** where the total number of paths examined is bounded, independent of the size of the database **54**, without a loss in effectiveness. To this end, we have an implementation of the algorithm **2044** such that a cluster link **2004** is defined recursively.

We define C_L (N_0, N_N), the order-L cluster link **2004** from start node **2008** to destination node **2008**, as the cluster link **2004** between N_0 and N_N, considering only paths of length less than or equal to L. Then, we can derive C_{L+1} (N_0, N_N) from C_L (N_0, N_N) and C_1 (N_0, N_N).

The assumption is that most of the paths P_L (N_0, N_N) of length L (or greater) from N_0 to N_N will not have a significant impact on cluster link generation. Therefore, we an use a set of candidate duster links **2004** C_L (N_0, N_N) as a summary of that path information for the purpose of determining C_{L+1} (N_0, N_N). This assumption has a significant impact on the performance of the algorithm **2044** in this implementation, since the search space is significantly reduced at each step. The computer processing "cost" of generating cluster links **2004** is bounded by the size of the candidate cluster link **2004** sets generated at the intermediate steps, rather than by the total number of relevant paths in the database **54**.

The size of the candidate duster link **2004** set generated at each intermediate step affects the speed of the algorithm **2044** in this implementation. If too many candidate cluster links **2004** are generated at each intermediate step, the algorithm **2044** is too slow. On the other hand, if too few candidate cluster links **2004** are generated, and too many paths are pruned, then C_L (N_0, N_N) is no longer an accurate summary of P_L (N_0, N_N).

Finally, since the weights **2034** of the individual candidate cluster links **2004** in C_L (N_0, N_N) are generally much greater than the weights **2034** of the individual paths in P_L (N_0, N_N), the damping factors D_N used to derive the combined weights **2034** at each step must be decreased accordingly in this implementation.

The specifics for the basic generator algorithm **2044** of this implementation, for determining the set of order N cluster links **2004** from a given start node **2008** N_0, are shown in FIG. **3**H. The generator algorithm **2044** works for any value of N greater than zero. If N=1, the set of candidate cluster links **2004** generated is simple. The processing cost of determining the candidate cluster links **2004** increases with N. In practice, N=3 appears to yield the best results.

The generator algorithm **2044** starts by initializing the candidate cluster link **2004** set **2048** and creating a loop for i=0 to N **2052**. The generator algorithm **2044** then performs a series of steps for each path P **2056**. First, it selects the destination node **2008** as the node to analyze and retrieves the set of direct links **2032** (L) from the selected node **2008** to any other node **2008** in the database **54**, N_{i+1}. Second, for each direct link **2032** L, the generator algorithm **2044** performs a series of steps.

The generator algorithm **2044** creates a new path P' of length i+1 consisting of the path P plus the direct link **2064** L from the selected node **2008** to the node **2008** N_{i+1} **2056**. The algorithm **2044** then determines the combined weight **2034** WC_{i+1} from WC_i, the weight **2034** of the path P, and W_{i+1}, the weight **2034** of Link **2004** L **2064**, using the following preferred formula:

$$WC_{i+1}=\min(WC_i, D_{i+1}*W_{i+1}).$$

Following these computations, the generator algorithm **2044** decides whether there already is a paths P in the cluster link **2004** from N_0 to N_{i+1} **2068**. If there is a not already a path, the algorithm **2044** adds P' to C_{i+1} **2072**. If there already is a path, the algorithm **2044** adds WC_{i+1} to the weight **2034** of the existing path in C_{i+1} **2076**. These steps are then repeated as necessary.

Once the candidate cluster link **2004** set has been generated, deriving the actual cluster links **2004** is a simple matter of selecting or choosing the T top rated candidate links **2004**, and eliminating the rest. In practice, the following formula has yielded good results:

$$T=\min(\text{constant}, 4*d),$$

where d is the number of direct links **2004** from N_0. Setting the constant equal to twenty has yielded good results. More than T cluster links **2004** may be generated if there are ties in the ratings. After each iteration, the candidate cluster link **2004** set C_i may be pruned so that it contains only the top candidate cluster links **2004** (for example, the top **200**).

FIGS. **4**A and **4**B are high level flow charts that illustrate the general flow of the subroutines of the CSPDM **66**. FIG. **4**A illustrates that the flow of various search routines depend on the type of search initiated by the researcher. The diagram further illustrates the interaction between the CSPDM **66** and the GUI Program **70**. FIG. **4**B illustrates the sequence of subroutines in the CSPDM **66** program and the user interactions with the subroutines. FIG. **4**B further shows that the researcher can access the different search subroutines and use information that the researcher has already received to find new information.

FIG. **4**B provides a high level flow chart illustrating the sequence of subroutines in the CSPDM **66** program and the researcher's interactions with the subroutines. Assuming that the database **54** the researcher desires to access has been proximity indexed, the researcher must log on **260** to the database **54**. By entering the appropriate information into the Computer Processor **30** via the input means, the researcher electronically accesses **264** the database **54** and enables the CSPDM **66** to search **200** the database **54**.

FIGS. **4**A and **4**B both show the preliminary options that the researcher can choose from before selecting one of the searching subroutines of the CSPDM **66**. The CSPDM **66** questions the researcher on whether the researcher has identified a pool of textual objects **204**. If the researcher has selected a pool of textual objects **204**, then the researcher is able to choose one of the pool search **208** subroutines **212**. If the researcher has not selected a pool of textual objects, the CSPDM **66** questions the researcher on whether the researcher has selected a single textual object **216**. If the researcher has selected a single textual object **216**, then the researcher is able to choose one **220** of the textual object searches **224**. If the researcher has not selected either a pool of textual objects **204** or a single textual object **216**, then the researcher must execute a Boolean Word Search or alternate Pool-Generation Method **228** to retrieve textual objects **268**, **272**.

After CSPDM **66** subroutine has executed a particular search, the CSPDM **66** retrieves the appropriate data from the database **54**, analyzes the data, and sends the data to the GUI Program **70** in order for the GUI Program **70** to display the results of the search on the display **38**.

FIG. **4**B illustrates that after the CSPDM **66** has completed the above procedure, the researcher has the option to exit the CSPDM **66** by logging off, executing a search based on the results of a previous search, or executing a new search.

FIGS. **4**A and **4**B also depict the seven subroutines of the CSPDM **66**. There are three textual object search subroutines **224** and four pool search subroutines **212**. The three textual object search subroutines **224** are: the Cases-In Subroutine **232**, the Cases-After Subroutine **236** and the Similar Cases Subroutine **240**. The four pool search subroutines **212** are the Pool-Similarity Subroutine **244**, the Pool-Paradigm Subroutine **248**, the Pool-Importance Subroutine **252**, and the Pool-Paradigm-Similarity Subroutine

256. Each of these subroutines are described in more detail in FIGS. **4**C to **4**I. The following is a step by step description of the subroutines **224**, **212** of the CSPDM **66**.

Section A Cases-In Subroutine **232**

FIG. **4**C is a high level flow chart for the Cases-In Subroutine **232**.

1. The researcher must select a single textual object **400**.

2. The researcher selects the Cases-In Subroutine **232** option.

3. The Cases-In Subroutine **232** examines the n×n Opinion Citation Matrix and other matrices **404** created by the Proximity Indexing Application Program **62** and retrieves the textual objects to which the selected textual object refers **408**, data relating to the number of times the selected textual object refers to the retrieved textual objects, data relating to the importance of each textual object, and other relevant data.

Section B Cases-After Subroutine **236**

FIG. **4**D is a high level flow chart for the Cases-After Subroutine **236**.

4. The researcher must select a single textual object **400**.

5. The researcher selects the Cases-After Subroutine **236** option.

6. The Cases-After Subroutine **236** examines the n×n Opinion Citation Matrix and other matrices **412** created by the Proximity Indexing Application Program **62** and retrieves the textual objects that refer to the selected textual object **416**, data relating to the number of times the retrieved textual objects refer to the selected textual object, data relating to the importance of each textual object, and other relevant data.

Section C Similar-Cases Subroutine **240**

FIG. **4**E is a high level flow chart for the Similar-Cases Subroutine **240**.

7. The researcher must select a single textual object **400**.

8. The researcher selects the Similar-Cases Subroutine **240** option,

9. The Similar-Cases Subroutine examines the q×q Section Similarity Matrix and other matrices **420** created by the Proximity Indexing Application Program **62** and retrieves the textual objects that are similar to the selected textual object **424**, data relating to the degree of similarity between the selected textual object and the retrieved textual objects, data relating to the importance of each textual object, and other relevant data. In order to be retrieved, a textual object must have a similarity coefficient with respect to the selected textual object of at least a minimum value. The preferred embodiment sets the minimum similarity coefficient of four percent (4%).

Section D Pool-Similarity Subroutine **244**

FIG. **4**F is a high level flow chart for the Pool-Similarity Subroutine **244**.

10. The researcher must select a pool of full textual objects **428**.

11. The researcher must then select a single full textual object **400** to which in compare the pool of full textual objects. It should be noted that the researcher can select the single textual object from the selected pool of textual objects, or the researcher can select a textual object from outside of the pool **432**.

12. The Pool-Similarity Subroutine **244** examines the n×n Opinion Similarity Matrix and other matrices **436** and values created by the Proximity Indexing Application Program **62** for the selected full textual object and the pool of full textual objects.

13. The Pool-Similarity Subroutine **244** determines the degree of similarity of other full textual objects in the pool to the selected full textual object **440**.

Section E Pool-Paradigm

FIG. **4**G is a high level flow chart for the Pool-Paradigm Subroutine **248**.

14. The researcher must select a pool of full textual objects **428**.

15. The Pool-Paradigm Subroutine **248** examines the n×n Opinion Proximity Matrix, the n×n Opinion Similarity Matrix and other matrices and values created by the Proximity Indexing Application Program **62** for the pool of full textual objects **448**.

16. The Pool-Paradigm Subroutine **248** determines the Paradigm full textual object by calculating the mean of the Euclidean distances of all the textual objects in the pool **452**.

17. The Pool-Paradigm Subroutine **248** determines the similarity of the other full textual objects in the pool to the Paradigm full textual object **456**.

Section F Pool-Importance Subroutine **252**

FIG. **4**H is a high level flow chart for the Pool-Importance Subroutine **252**.

18. The researcher must select a pool of full textual objects **428**.

19. The Pool-Importance Subroutine **252** examines **448** the n×n Opinion Citation Matrix, the n×n Opinion Similarity Matrix, numerical factors and other matrices and values created by the Proximity Indexing Application Program **62** for the pool of full textual objects **460**.

20. The Pool-Importance Subroutine **252** then ranks the importance of each of the full textual objects in the pool **464**.

FIG. **4**I is a high level flow chart showing two possible alternate Pool-Paradigm-Similarity Subroutines **256**.

Section G Pool-Paradigm-Similarity Subroutine **256** (Option 1) **256**

21. The researcher must select a pool of k full textual objects where k equals the number of full textual objects in the pool **428**.

22. For each of the k full textual objects, the Pool-Paradigm-Similarity Subroutine **256** selects a n×1 vector from the corresponding column of the n×n **468**.

23. The Pool-Paradigm-Similarity Subroutine **256** creates an n×k matrix by grouping the n×1 vector representing each of the k full textual objects beside each other.

24. The Pool-Paradigm-Similarity Subroutine **256** calculates the mean of each row of the n×k matrix and enters the mean in the corresponding row of an n×1 Paradigm Proximity Vector **472**.

25. The Pool-Paradigm-Similarity Subroutine **256** combines the n×1 Paradigm Proximity Vector with the n×n Opinion Proximity Matrix to create an (n+1)×(n+1) Paradigm Proximity Matrix **476**.

26. From the (n+1)×(n+1) Paradigm Proximity Matrix, the Pool-Paradigm-Similarity Subroutine **256** evaluates the Euclidian distances and empirical data to create an (n+1)×(n+1) Paradigm Similarity Matrix **480**.

27. The Pool-Paradigm Similarity Subroutine **256** searches the row in the (n+1)×(n+1) Paradigm Similarity Matrix that corresponds to the Paradigm full textual object and retrieves the full textual objects that have a maximum degree of similarity with the Paradigm full textual object **500**.

Section H Pool-Paradigm Similarity Subroutine **256** (Option 2)

28. The researcher must select a pool of k full textual objects where k equals the number of full textual objects in the pool **428**.

29. For each of the k full textual objects, the Pool-Paradigm-Similarity Subroutine **256** selects an n×1 vector from the corresponding column of the n×n **484**.

30. The Pool-Paradigm-Similarity Subroutine **256** creates an n×k matrix by grouping the n×1 vector for each of the k full textual objects beside each other.

31. The Pool-Paradigm-Similarity Subroutine **256** calculates the mean of each row of the n×k matrix and enters the mean in the corresponding row of an n×1 Paradigm Pattern Vector PF **488**.

32. The Pool-Paradigm-Similarity Subroutine **256** combines the n×1 Paradigm Pattern Vector PF with the n×n Opinion Pattern Matrix to create a (n+1)×(n+1) Paradigm Pattern Matrix **492**.

33. From the (n+1)×(n+1) Paradigm Pattern Matrix, the Pool-Paradigm-Similarity Subroutine **256** evaluates the Euclidean distances between the rows of the Paradigm Pattern Matrix and creates an (n+1)×(n+1) Paradigm Proximity Matrix **496**.

34. From the (n+1)×(n+1) Proximity Matrix, the Pool-Paradigm-Similarity Subroutine **256** evaluates the Euclidean distances between the rows of the (n×1)×(n×1) Paradigm Proximity Matrix and empirical data to create an (n+1)×(n+1) Paradigm Similarity Matrix **480**.

35. The Pool-Paradigm Similarity Subroutine **256** searches the row in the (n+1)×(n+1) Paradigm Similarity Matrix that corresponds to the Paradigm full textual object and retrieves the full textual objects that have a minimum degree of similarity with the Paradigm full textual object **500**.

Application of the Proximity Indexing Technique

The above Proximity Indexing Application Program **62** and CSPDM **66** have a number of different applications and versions. Three of the most useful applications are described below.

The first type of Proximity Indexing Application Programs **62** is for use on very large databases. The matrices generated by this type of Proximity Indexer are "attached" to the database **54**, along with certain clustering information, so that the database **54** can be searched and accessed using the Cases-In Subroutine **232**, Cases-After Subroutine **236**, Similar Cases Subroutine **240**, Pool-Similarity Subroutine **244**, Pool-Paradigm Subroutine **248**, Pool-Importance Subroutine **252** and Pool-Paradigm-Similarity Subroutine **256** of the CSPDM **66**.

The second type of Proximity Indexing Application Program **62** is a Proximity Indexer that law firms, businesses, government agencies, etc. can use to Proximity Index their own documents in their own databases **54**. The researcher can navigate through the small business's preexisting database **54** using the Cases-In Subroutine **232**, Cases-After Subroutine **236**, Similar Cases Subroutine **240**, Pool-Similarity Subroutine **244**, Pool-Paradigm Subroutine **248**, Pool-Importance Subroutine **252** and Pool-Paradigm-Similarity Subroutine **256** of the CSPDM **66**. In addition, this type of Proximity Indexer Application Program **62** will be designed to be compatible with the commercial third-party databases **54** which are Proximity Indexed using the first type of program. In other words, the researcher in a small business may "weave" in-house documents into a commercial database **54** provided by a third party, so that searches in the large database **54** will automatically bring up any relevant in-house documents, and vice versa.

The third type of Proximity Indexing Application Program **62** involves the capacity to do Proximity indexing of shapes. Each image or diagram will be treated as a "textual object." The various matrix coefficients can be generated purely from topological analysis of the object itself, or from accompanying textual information about the object, or from a weighted combination of the two. The text is analyzed using the Proximity Indexing Application Program **62** as explained above. Shapes are analyzed according to a coordinate mapping procedure similar to that used in Optical Character Recognition ("OCR"). The numerical "maps" resulting from scanning the images are treated as "textual objects" that can be compared through an analogous weighing algorithm to generate a proximity matrix for every ordered pair of "textual objects" in the database **54**. A similarity matrix can then be generated for each ordered pair, and the results organized analogous to a database **54** totally comprised of actual text.

This third type of Proximity Indexing Applications Program **62** can provide "Proximity Indexed" organization access to many different types of objects. For example, it can be used to search patent diagrams, or compare line drawings of known pottery to a newly discovered archeological find. It can be used to scan through and compare police composite drawings, while simultaneously scanning for similar partial descriptions of suspects. It can be used to locate diagrams of molecular structures, appraise furniture by comparing a new item to a database **54** of past sales, identify biological specimens, etc., etc.

FIG. **5**A is a high level drawing that depicts one embodiment of the GUI Program **70** and its interaction with both the CSPDM **66** and the display **38**. The GUI Program **70** has one or more display subroutines. One embodiment contains seven display subroutines. The seven subroutines comprise three textual object display subroutines **504** and four pool display subroutines **508**. The three textual object display subroutines **504** are the Cases-In Display Subroutine (CIDS) **512**, the Cases-After Display Subroutine (CADS) **516** and the Similar-Cases Display Subroutine (SCDS) **520**. The four pool display subroutines **508** are the Pool-Similarity Display Subroutine (PSDS) **524**, the Pool-Paradigm Display Subroutine (PPDS) **528**, the Pool-Importance Display Subroutine (PIDS) **532** and the Pool-Paradigm-Similarity Display Subroutine (PPSDS) **536**. The three textual object display subroutines **504** receive data from the corresponding textual object search subroutine **224** of the CSPDM **66**. Similarly, the four pool display subroutines **508** receive data from the corresponding pool search subroutine **212** of the CSPDM **66**. Once the display subroutines have processed the data received by the search subroutines, the data is sent to the integrator **540**. The integrator **540** prepares the data to be displayed in the proper format on the display **38**.

FIGS. **5**B through **5**H depict screens generated by the textual object display subroutines, CIDS **512**, CADS **516** and SCDS **520**. The three types of screens are the Cases In screen **1000**, the Cases After screen **1004** and the Similarity Screen **1008**, respectively. The Similarity Screen **1008** provides the most "intelligent" information, but all three screens generated by the textual object display subroutines **504** work in tandem as a system. The other screens created by the pool display subroutines are variances of these three, and also work in tandem with each other and with the three textual object display screens.

FIG. **5**B depicts the "Cases After" **1004** Screen created by the CADS **516** for the textual object, *Terry* v. *Ohio,* 392 U.S. 1 (1968). The Cases-After subroutine **236** search produces all of the textual objects. in the designated field (here D.C. Circuit criminal cases since 1990) that cite *Terry.* The number "12" **1080** in the upper left hand corner indicates that there are a total of 12 such textual objects. The vertical axis **1012** indicates the degree to which a given textual object relied upon *Terry.* The number "10" immediately below the 12 indicates that the textual object in the field which most relied upon *Terry* namely *U.S.* v. *Tavolacci,* 895

F.2d 1423 (D.C. Cir. 1990), discusses or refers to *Terry* in ten of its paragraphs.

The Tear-Off Window **1016** feature is illustrated in FIG. **5**B by the Tear-Off Window **1016** for *U.S.* V. *McCrory,* 930 F.2d 63 (D.C. Cir. 1991). The four Tear-Off Window active boxes **1020** (displayed on the Tear-Off Window **1016**): 1) open up the full text **1104** of *McCrory* to the first paragraph that cites *Terry;* 2) run any of the three searches, namely Cases-In Subroutine **232** Cases-After Subroutine **236** or similar cases Subroutine **240** for *McCrory* itself (the default is to run the same type of search, namely Cases-After Subroutine **236** again); 3) hide the *Terry* execute search window **1024**; and 4) bring the *Terry* Execute Search window to the foreground, respectively. The weight numeral **1028** indicates the number of paragraphs in *McCrory* that discusses or refers to *Terry,* in this textual object (in this example there is only one).

The Cases After screen **1004** for a given Textual object B displays a Textual Object Active Box **1032** representing every subsequent textual object in the database **54** that refers explicitly to Textual object B. The analysis starts with the same pool of material as a Shepards™ list for Textual object B. As well as some additional material not gathered by Shepards. However, the Cases After screen **1004** conveys a wealth of information not conveyed by a Shepards™ list.

The horizontal axis **1036** may represent time, importance or any other means of measurement to rank the textual objects. The Shepards list itself contains no information as to when a case was decided. The vertical axis **1012** similarly may represent any means of measurement to rank the textual objects. In the preferred embodiment, the vertical axis **1012** represents the degree to which the subsequent Textual object C relied upon the original Textual object B. The display **38** makes it obvious when a textual object has received extensive discussion in another textual object, or provides key precedent for a subsequent textual object, or merely mentions the earlier textual object in passing. It also provides guidance as to possible gradations in between extensive, or merely citing.

The "shape" of the overall pattern of active boxes on the Cases After screen **1004** provides a rich lode of information to be investigated. For example, a "dip" in citation frequency immediately after a particular textual object suggests that the particular textual object, while not formally overruling Textual object B, has largely superseded it. A sudden surge in citation frequency after a particular Supreme Court case may indicate that the Supreme Court has "picked up" and adopted the doctrine first enunciated in Textual object B. The researcher can instantly determine if the holding of Textual object B has been adopted in some circuits but not in others, if Textual object B is losing strength as a source of controlling precedent, etc. None of this information is now available to lawyers in graphical or any other form.

As with the Cases In screen **1000**, every Textual Object Active Box **1032** on the Cases After screen **1004** is active, and includes a Tear-Off Window **1016** that may be moved by dragging on the tear-off window **1016** with a mouse **42**, and that tear-off window **1016** becomes a text Tear-Off Window **1040**, visible even when one moves on to other searches and other screens. Thus one may "tear off" for later examination every relevant citation to Textual object B, or even for a group of textual objects. The text tear-off windows **1040** "tile"; that is, they can be stacked on top of one another to take up less room. There is also a "Select All" feature (not shown), that creates a file containing the citations of every textual object retrieved in a given search.

In Cases After screen **1004** mode, clicking on the expanded-view button **1044** of the text tear-off window **1040** opens the text of the subsequent Textual object C to the first place where Textual object B is cited. A paragraph window **1048** displays a paragraph selection box **1052** indicating what paragraph in Textual object C the researcher is reading, and a total paragraph box **1056** indication how many paragraphs Textual object C contains in total. The user can view paragraphs sequentially simply by scrolling through them, or see any paragraph immediately by typing its number in the paragraph selection box **1052**. Clicking on a Next paragraph active box **1060** immediately takes the researcher to the next paragraph in Textual object C where Textual object B is mentioned. Traditional Shepardizing allows the researcher to explore the subsequent application of a doctrine in a range of different factual situations, situations that help to define the outer contours of the applicability of a rule. Combining the expanded-view button **1044** functions and "Next Paragraph" active box **1060** functions allows the researcher to study how Textual object B has been used in all subsequent textual objects, in a fraction of the time the same task currently requires with available searching methods.

Perhaps the most fundamental form of legal research is "Shepardizing." A researcher starts with a textual object known to be relevant, "Textual object B," and locates the "Shepards" for that textual object. The "Shepards" is a list of every subsequent textual object that explicitly refers to Textual object B. The researcher then looks at every single textual object on the list. Shepardizing is often painstaking work. Many subsequent references are made in passing and have almost no legal significance. Although Shepards includes some codes next to its long lists of citations, such as "f" for "followed" and "o" for "overruled," the experience of most lawyers is that such letters cannot be relied upon. For example, the researcher may be citing Textual object B for a different holding than that recognized by the anonymous Shepards reader, interpreting Textual object B differently, or interpreting the subsequent textual object differently. However, for really thorough research, checking a Shepards type of list is essential. The researcher must make absolutely sure that any textual object cited as legal authority in a brief, for instance, has not been superseded by later changes in the law.

Very often, textual objects located on the Shepards list for Textual object B refer back to other important textual objects, some of which may predate Textual object B, all of which may be Shepardized in turn. This "zig-zag" method of research is widely recognized as the only way to be sure that one has considered the full line of textual objects developing and interpreting a doctrine. The real power of the Cases After screen **1004** emerges when it is used in conjunction with the Cases In screens **1000** and Similarity screens **1008**. Using the preferred embodiment, the researcher may engage in the same kind of careful "zig-zag" study of a legal doctrine in a much more efficient manner.

For example, consider the following hypothetical search. The researcher reads Textual object B, and makes a list of every Supreme Court textual object it substantially relies upon, perhaps six textual objects. The researcher then Shepardizes Textual object B and reads each of those textual objects, in order to find other Supreme Court textual objects that they relied upon, perhaps eight. One then Shepardizes those fourteen Supreme Court decisions, in order to find any Court of Appeals cases in a selected circuit within the last three years on the same basic topic. This process would take at least an hour, even using Shepards through an on-line service. The same search can be performed with the present invention using the Cases In screens **1000** and Cases After screens **1004** in under five minutes.

In order to perform the same search, a researcher can pull up both the Cases In screens **1000** and Cases After screens **1004** for Textual object B simultaneously. The researcher can then "tear-off" all of the Supreme Court Cases on both lists, run Cases-After Subroutine **236** searches on every Supreme Court Case mentioned on either list, then examine the Cases In screens **1000** for all of the Supreme Court cases produced by these searches. The researcher can locate every recent Court of Appeals case from a selected circuit mentioned in any of those Supreme Court cases. Use of the Similarity screen **1008** as well, allows the researcher to find the pool of relevant Court of Appeals full textual objects even faster.

FIG. **5**C depicts the Cases After Screen **1004** for *U.S.* v. *Lam Kwong-Wah,* 924 F.2d 298 (D.C. Cir. 1991). FIG. **5**C shows a text Tear-Off Window **1040** on a Cases After Screen **1004**, (in this textual object the Tear-Off Window **1016** for *U.S.* v. *Barry,* 938 F.2d 1327 (D.C. Cir. 1991), is opened using the full text active box **1064**. A text Tear-Off Window **1040** containing the text of *Barry* opens, to the first cite of *U.S.* v. *Lam Kwong-Wah* at paragraph **15**. Clicking on the Next Paragraph active box **1060** will open the text of *Barry* to the next paragraph that cites *Lam Kwong-Wah.*

The number "34" in the lower-left corner of the total paragraph box **1056** indicates that Barry has a total of **84** paragraphs in the cite *U.S.* v. *Lam Kwong-Wah.* Dragging the small squares **1068** to the left and below the text allow the researcher to move within a paragraph, and from paragraph to paragraph, in the text of *Barry,* respectively. The empty space below the text **1072** would contain the text of any footnote in paragraph 15. The compress window active box **1074** now closes the window and replaces it with the corresponding Textual Object Active Box **1032**.

FIG. **5**D depicts the Cases In Screen **1000** for *U.S.* v. *North,* 910 F.2d 843 (D.C. Cir. 1990). FIG. **5**D contains a Textual Object Active Box **1032** representing every textual object or node with persuasive authority, cited in the text of *North.* The vertical axis **1012** represents the degree to which *North* relied upon a given textual object. In this example it is immediately apparent that *Kastigar* v. *United States* 406 U.S. 441 (1972) is the most important precedent, and its Tear-Off Window **1016** have been activated. The weight numeral **1028** indicates that *Kastigar* is referred to in 77 paragraphs of *North.*

A highlighted Textual Object Active Box **1076** can be created by clicking on it, as has been done with *U.S.* v. *Lily,* 651 F.2d 611. The number "212" in the case number box **1080** indicates that citations to two-hundred-twelve distinct texts appear in *North.* Fewer are visible because the textual object active boxes **1032** "tile" on top of one another; the "Zoom" feature is used to focus on a smaller area of the screen, and ultimately resolves down to a day-by-day level, making all the textual object active boxes **1032** visible.

The unique Cases In screen **1000** provides a schematic representation of the precedent from which Textual object A is built. The Cases In screen **1000** contains a textual object active box **1032** representing every textual object which is relied upon, or even mentioned, in Textual object A. Any citation in textual object A to a textual object that possesses potential persuasive authority, whether a statute, constitutional provision, treatise, scholarly article, Rule of Procedure, etc., is treated as a "textual object." The textual object active boxes **1032** are color-coded to indicate the court or other source of each textual object. Supreme Court cases are red, Court of Appeals cases are green, District Court cases are blue, and statutes are purple, for example. Each Textual Object Active Box **1032** contains the full official citation **1084** of its textual object. Clicking on any Textual Object Active Box **1032** immediately pulls up a larger window, known as a tear-off window **1016**, also containing the full citation **1084** to the textual object (Tear-Off Window Citation **1088**), its date **1092**, its circuit **1096**, and its weight numeral **1028** to the textual object being analyzed. The user may then drag the Tear-Off Window **1016** free of the Textual Object Active Box **1032** and release it.

This creates a text Tear-Off Window **1040** that remains visible until the researcher chooses to close it, no matter how many subsequent screens the researcher examines. The text Tear-Off Window **1040** can be moved anywhere by dragging it with the mouse **42**. The text Tear-Off Window **1040** contains small text active boxes **1100** allowing the researcher to access or "pull up" the full text **1104** of the textual object it represents with a single click of the mouse **42**. This feature also allows the researcher to run Cases-In Subroutine **232**, Cases-After Subroutine **236**, and Similar Cases Subroutine **240** searches on the textual object. (See below for a description of the Similarity screen **1008**).

The organization of the boxes on the screen, including their position on the horizontal axis **1036** and vertical axis **1012**, represents the real "intelligence" behind the Cases-In screen **1000**. The horizontal axis **1036** in the preferred embodiment represents time, with the left margin **1108** corresponding to the present, i.e., the date 1992 when the search is run. The right margin **1112** represents the date of decision of the earliest textual object cited in Textual object A. (Certain special materials, such as treatises updated annually, and the U.S. Constitution, are located in a column **1116** to the left of the margin.)

The vertical axis **1012** in the preferred embodiment represents the degree to which Textual object A relied upon each particular textual object it contains. For example, if the Cases In screen **1000** is run on a district court case (Textual object A) which happens to be a "stop and search" textual object that mainly relies upon *Terry* v. *Ohio,* 392 U.S. 1 (1968), *Terry* will be at the top of the screen, with all other textual object active boxes **1032** appearing far below. The researcher can thus access the text of *Terry* directly without ever reading the text of Textual object A. Of course, the full text **1104** of Textual object A is also instantly available if desired. If the researcher wants to see where *Terry* "came from," the researchers can instantly, by clicking on a text active box **1100** within the *Terry* text Tear-Off Window **1040**, run the Cases-In Subroutine **232** for *Terry*—and so on. There is no limit to the number of "levels" or "generations" the researchers may explore using this technique. It is therefore possible (assuming a sufficient database **54**) to find, in a matter of seconds, without having to read through layers of texts, the possibly long-forgotten eighteenth-century precursors to a modern doctrine.

The Cases In screen **1000** creates an instant visual summary or "blueprint" of a textual object. The blueprint can help a researcher make a preliminary judgment about whether a particular textual object is worth closer examination. Viewing the Cases In screens **1000** for a group of textual objects allows a researcher to recognize whether there are precedents common to that group. The blueprint tells the researcher whether Textual object A is primarily a statutory construction case, a textual object that relies on local Court of Appeals cases without Supreme Court support, a textual object relying on precedent outside the circuit **1096** as persuasive authority, etc.

The initial Cases In screen **1000** presents every citation within a given textual object. In a textual object with an

unusually large number of citations, the screen will be crowded with textual object active boxes **1032**. The GUI therefore contains a "zoom" feature that allows the researcher to expand any small portion of the screen. To get back to the "big picture," the researcher simply selects the "Fit in Window" menu item, or else selects the "zoom out" feature. The same "zoom," "zoom out," and "Fit in Window" functions are present in the Cases After screen **1004** and Similarity screen **1008** as well.

The routine that calculates "degree to which Textual object A relies upon the cited textual object" clearly ranks major textual objects at the top, textual objects mentioned only in passing at the bottom, and textual objects of potentially greater relevance in between via display the appropriate textual object active boxes **1032** in the appropriate place. In addition, the routine can recognize when a highly relevant textual object is mentioned only in passing and give a higher weight to that textual object than it would otherwise receive in the ranking procedure.

The "intelligence" behind the entire GUI is driven by the knowledge that the lawyers do not want the computer to do legal analysis or make judgments for them, but simply guide them through the great mass of irrelevant material to those texts where lawyerly analysis of a problem begins.

The Cases In screen **1000** is designed with practical legal research in mind. It is common in legal research to locate a lower court textual object on the correct topic, call it "local Textual object A." However, the researcher desired to find the most persuasive authority available. The aim of this type of research is to find the "lead" textual object or textual objects on a particular topic. The researcher ultimately desires the first textual object, most famous textual object, and most recent textual objects of the Supreme Court (or state Supreme Court in state law issues) that stand for the same principle. ("Lead" textual objects also occur at the intermediate and trial court level.)

The standard way to find lead textual objects is to read through the text of a local Textual object A until one finds references to "higher court textual objects," then look up each of those higher court textual objects in turn. The researcher then reads the text of those textual objects until the researcher determines the textual objects they have in common, the textual objects that appear many times. Very often, the lower court textual object from which the researcher started is of no real value in and of itself—it may well be from a different local jurisdiction—and the researcher reads through it only to find citations within it. Since the GUI quickly locates and schematically diagrams the textual objects, this process is accelerated dramatically using the GUI.

FIGS. **5**E through **5**G depict multiple Similar Case Subroutine **240** searches run in sequence. A Similarity Screen **1008** for *U.S.* v. *Caballero* 936 F.2d 1292 (D.C. Cir. 1991), reveals via the case number box **1080**, that 17 textual objects were retrieved by Similar Cases Subroutine **240** search. The vertical axis **1012** indicates that the textual objects retrieved had similarity coefficients **1120** between 4% and 15% with respect to *U.S.* v. *Caballero.* Textual objects with less than 4% similarity are not shown. The vertical axis **1012** represents degree of similarity, or topical relatedness, so that 100% would be two identical texts. The Tear-Off Window **1016** of *U.S.* v. *Nurse,* 916 F.2d 20 (D.C. Cir. 1990) shows that the textual object has a similarity of 9%.

The Similarity screen for a given Textual object C is organized like the Cases In screen **1000** and Cases After screen **1004**, with the same color-coded textual object active boxes representing textual objects, and time on the horizontal axis **1036**. However, the vertical axis **1012** represents the degree to which the represented textual object is related to Textual object C. The system is built on the principle that legal doctrines tend to emerge out of lines of textual objects developing a legal principle. Lines of textual objects contain "lead" textual objects that establish basic rules and subsequent textual objects that do not establish new rules, but apply and re-interpret the pre-existing rules in various circumstances. Some lead textual objects invent new doctrines, while others modify or redirect the law based on earlier precedent.

The routine that operates behind the Similarity screen **1008** determines which line or lines of textual objects that Textual object C can be grouped. The routine then ranks the textual objects in that line depending on how closely they are related to Textual object C. For example, a typical similarity search starting with a Court of Appeals case in a certain circuit, Textual object D, will find the Supreme Court and Court of Appeals cases that have established the principles followed in Textual object D. The Supreme Court and Court of Appeals case will appear as textual object active boxes whether or not they are cited in Textual object D. Furthermore, the Similar Cases Subroutine **240** search will find the textual objects decided subsequent to Textual object D that have applied, and possibly modified, those principles, whether or not those textual objects cite Textual object D.

Similarity searches allow a researcher to find textual objects on the same topic that do not share common phrases and might be overlooked by a Boolean word search. Similarity searches also allow researchers, who only have an obscure district court case, to "tap in" to the lead textual objects in any area. By organizing all case law in "conceptual space," the Similarity screens **1008** allow one to locate emerging topics that have not been formally recognized by those assigning "key numbers" or otherwise manually classifying textual objects—or even by the authors of the textual objects themselves.

The "shape" of a Similarity Screen **1008** may convey a great deal of information about a particular legal concept. For example, the screen conveys to the researcher whether a certain concept, which is essentially novel, is supported by Supreme Court case law. Or is an old doctrine that has been recently applied in a new context. The system as a whole gives lawyers the ability to assess what textual objects are "available" on their topic, and to zero in on the textual objects that are most useful. The researcher has the ability to track down every subsequent reference to any particular textual objects by utilizing multiple "Cases After" searches, identifying core precedents through "Cases In" searches, and by running new "Similarity" searches to obtain any textual objects that emerge in closely related topic areas. The "Similarity" algorithm is more "aggressive" then the others, since it contains built-in judgments as to what "relatedness" means. It also judges what is no longer sufficient to display on the screen. The bottom edge of the screen represents a minimum degree of similarity below which the connections are too tenuous to be worth pursuing. In the commercial product, this minimum level can be reset at the preference of the user.

FIG. **5**F is the Similarity Screen **1008** for *U.S.* v. *Nurse.* Clicking on the run search Tear-Off Window active box **1128**, which is on the Tear-Off Window **1016** for Nurse produces FIG. **5**F. Clicking on the Textual Object Active Box **1032** for *U.S.* v. *Jordan,* 951 F.2d 1278 (D.C. Cir. 1991) long enough to pull up its Tear-Off Window **1016**, and then clicking on *Jordan's* run search Tear-Off Window active box **1020** (not shown), produces the Similarity Screen **1008** shown in FIG. **5**G.

FIG. **5**G shows how multiple tear-off windows **1016** can be shown at the same time, here the *U.S.* v. *Jordan* similarity Tear-Off Window **1016** depicts for the three textual objects most similar to *Jordan.* Note that *U.S.* v. *Jordan.* 958 F.2d 1085 (D.C. Cir. 1992), is very closely related, i.e., 41%, to *U.S.* v. *Jordan,* 951 F.2d 1278 (D.C. Cir. 1991), apparently as it is a subsequent full textual object decision of the same dispute as the first textual object.

FIG. **5**H depicts a close-up view of an Execute Search Window **1024**. The researcher can input a selected textual object that is either represented or not represented on a display **38** screen as a Textual Object Active Box **1032**. The researcher can title his search by inputing the title in the Title Search box **1132**. The researcher can then input the reference to the selected textual object in the reference input boxes **1136**. The reference input boxes of the preferred embodiment allow the researcher to refer to the selected textual object by Volume, Category, Page and/or Section by inputing the appropriate values in the volume reference box, category reference box, page reference box, and/or section reference box, respectively.

The researcher can also identify the type of search to be performed on the selected textual object by selecting the appropriate search in the Analysis box.

Once the researcher has inputed all the appropriate values, the researcher executes the search by activating the execute search button.

Referring generally to FIGS. **5**A through **5**H, the PSDS **524**, PPDS **528**, PIDS **532** and PPSDS **536** of the GUI Program **70**, also create similar displays to the CIDS **512**, CADS **516**, and SCDS **520** subroutines. The only major difference between the screens created by the three textual object display subroutines and the four pool display subroutines is the information contained in the Execute Search window and the options available in the analysis box.

The options in the analysis box enable a researcher to select a textual object outside the pool of textual objects and compare how the selected textual object relates to the pool of textual objects by selecting to the Pool-Similarity Subroutine **244**, the Pool-Paradigm Subroutine **248** or Pool-Importance Subroutine **252** of the CSPDM **66**.

The PSDS **524** creates a Pool-Similarity Screen **1008**. The vertical axis **1012** ranks the similarity of the objects in a pool of textual objects with respect to a selected textual object. All of the other aspects of this display **38** are similar to the Similar Cases Screen.

PPDS **528** creates a Pool-Paradigm Screen. The vertical axis **1012** ranks the similarity of the pool of textual objects on the screen with respect to the paradigm textual object. The paradigm textual object is calculated by averaging the mean of all the Euclidean distances of the pool of textual objects on the screen. All of the other aspects of this display **38** are similar to the Similar-Cases Screen.

The PIDS **532** creates a Pool-Importance Screen. The vertical axis **1012** ranks the importance of the pool of textual objects on the screen. All other aspects of the PIDS **532** display **38** are similar to the Cases-In Screen **1000** and Cases-After Screen **1004**.

The PPSDS **536** creates a Pool-Paradigm Similarity Screen **1008**. The vertical axis **1012** represents the similarity of all textual objects in the database **54** to the paradigm textual object created by a selected pool of textual objects. All other aspects of the PPSDS **536** display **38** are similar to Similar-Cases Screen **1008**.

Before displaying the text boxes **1032** representing result nodes **2104** on the screen to the user, the graphical user interface program **70** optimally organizes and arranges the location of text boxes **1032** on the X and Y axis. In the preferred embodiment, the GUI Program, **70** uses a layout of boxes algorithm to optimally place boxes within a window.

Referring to FIG. **7**, generally, a layout algorithm plots text boxes **1032** on a cartesian axis as determined by their X and Y values **1200**. The algorithm compares the locations of boxes **1032** within a display window to determine if there are any overlapping boxes **1204**. In order to perform this comparison, the preferred algorithm initializes a first loop, for i=0 to N, and chooses box_i to begin the comparison. The algorithm next creates a second loop, for j=1 to N, and chooses box_j to compare with box_i. For both loops, N is the number of result nodes obtained. Performing N^2 comparisons provides the optimal number of comparisons needed to determine the existence of any overlaps. If the boxes **1032** are a known size, certain steps may be eliminated from the method.

More particularly, a preferred algorithm compares the X and Y values of a first and second box **1032** to determine if the boxes **1032** are occupying the same Cartesian space **1204**. This comparison is accomplished by identifying the X and Y coordinate pairs of the corners of the two boxes **1032**, and then choosing one of the coordinate pairs of a corner of the second box **1032** to be compared. The X value of that pair is compared to the X values of all of the coordinate pairs of the corners of the first box **1032**. If the X value of the chosen coordinate pair is a value less than all of the first box corner X values or a value greater than all of the first box corner X values, then the algorithm compares the Y value of the chosen pair to all of the first box corner Y values. If the Y value of the chosen pair is either a value less than all of the first box corner Y values or a value greater than all of the first box corner Y values of the first box **1032**, then the algorithm determines that the boxes **1032** do not overlap. The algorithm adds 1 to counter j and then repeats the routine. The routine is repeated until j reaches N and then 1 is added to the i value, and the entire process is repeated again. This particular method ensures that every box is compared with every other box **1032**.

If during the comparison of the X values the algorithm finds that the X value of the chosen pair is greater than one of the first box corner X values, but is less than one of the first box corner X values, then the algorithm determines that the boxes **1032** overlap. If during the comparison of the Y values the algorithm finds that the Y value of the chosen pair is greater than one of the first box corner Y values but is less than one of the first box corner Y values, then the algorithm determines that the boxes **1032** overlap.

If the preferred algorithm has determined that two boxes **1032** overlap, then the algorithm moves **1208** one of the boxes **1032**. Preferably, this is accomplished by adjusting the Y values of the second box **1032** by increasing or adding a predetermined value (to its Y values.) The algorithm performs the above comparison routine **1204** again to see if there is an overlap. If there is an overlap, it moves **1208** the box again. Preferably, it adjusts **1208** the second box's Y value again, and compares **1204** again. If there is no overlap, then the algorithm adds 1 to the j counter and repeats the comparison routine with another box **1032** until N^2 comparisons have been completed.

When the preferred layout algorithm has ensured that no boxes **1032** overlaps, the algorithm determines whether the results of the search win fit on one screen **1212**. The algorithm compares the Y values of each of the boxes **1032** with the highest Y value represented on a single screen display. If the Y value of one or more boxes **1032** exceeds the highest Y value represented on the screen display, then

the preferred algorithm increases the length of the X axis and rescales the Y axis to match (e.g., doubling the length of the X axis) **1216**. The algorithm again compares the Y values of each of the boxes **1032** to the highest Y value on the screen to determine if the search results will fit on one screen **1212**. If they will not, then the algorithm adjusts the X axis **1216** again and compares the Y values **1212** again until the search results fit on one screen.

Once the search results fit on one screen, the algorithm replots **1200** all of the boxes **1032** to their coordinate positions, and then performs the overlap comparison check **1204** again to see if any boxes **1032** are overlapping. If boxes **1032** are overlapping, the algorithm performs its adjustment step **1208** and the axis resizing step **1216** until the window displays **1220** all of the result nodes on a single screen without any of the boxes overlapping.

At the option of the user, the algorithm can allow the display **1220** to scroll off the screen in the Y direction or the X direction without resizing **1216** the axes. This option enhances the information content of the map by keeping the scale of the axes small.

The preferred algorithm can perform this routine by adjusting **1208** the X axis, the Y axis, or both axes. The algorithm has the additional capability of graphically breaking an axis, if one or a few result nodes **2104** are so far away graphically from the main body of result nodes **2104** that representing the far away result nodes **2104** would unnecessarily encumber the graphical display of the main nodes. This graphical break may be represented by a squiggly line at the break point in the axis. Using this axis break allows all of the result nodes **2104** to be displayed on one window, and still maximizes the informational content that the relative spacing on the X or Y axis provides for the result nodes **2104** which are positioned closer together.

Various other specific methods of optimally organizing and locating boxes **1032** on a graphical computer display **38** may be used with the GUI Program **70**.

In the preferred embodiment, the graphical user interface **70** maximizes the types and quantity of information about particular boxes **1032**, nodes **2008**, objects in the database **54** that can be displayed without visually overloading the user. The preferred embodiments ergonomically and efficiently represent complex data sets. Each embodiment must strike a balance between that which is technically and intellectually possible to be displayed on a screen, and that which can be visually understood and comprehended by the typical user of the database **54** (on a screen). This method can also be used to display objects retrieved from a network, but it is not the preferred method at the present time.

An important feature of the invention is its use of a three-dimensional box to communicate information to the user, in addition to the information provided by the location of the box in the X and Y coordinates. Referring to FIG. **8**, in the preferred embodiment a three coordinate view or map is displayed on a two dimensional CRT screen. The variables represented by the X, Y, and Z coordinate planes may be interchanged from one coordinate to another. In other words, the X, Y or Z coordinate plane may represent, for example, the variable time. For use of the Z coordinate, it is preferred that a six-sided box **1033** be used and appear to be “floating” at its appropriate location in the Z direction. More importantly, the invention also can use the depth of the box **1032** (size of the box **1032** in the Z direction) to convey additional information to the user (in addition to the information provided by the location of the box **1032** in the X, Y, Z coordinate).

First, using box depth, a bit of binary information is passed along to the user by the fact that the box **1033** has no depth (little or nominal depth) or the box **1033** has a significant depth in the Z direction. In the preferred embodiment, this binary piece of information informs the user of whether or not there is available (hidden) data associated with that box **1033**. For example, a box **1033** or node which represents an object in the database **54** may have associated graphics, maps, menus, or text which is not shown. If the box **1033** is shown on the screen as having a significant depth then additional data associated with that box **1033** is available for viewing by the user. If the box **1033** has nominal or no depth then there is no additional data available to the user.

In addition to the binary information of whether or not additional data is available to the user, in the preferred embodiment the magnitude of the depth of the box **1033** corresponds to the amount of additional or hidden data available to the user. For example, if the box **1033** represents an object in the database **54** which has an extensive amount of associated data the magnitude of the depth of the box **1033** would be large in comparison to other boxes **1033** on the same screen. In this manner, a box **1033** which represents, for example, a textual object of great length would have a larger depth than a box **1033** representing a textual object with little or no text associated with that object in the data base. In this manner, important information is visually passed to the user easily and on the same screen on which other information about the database **54** is being presented.

Also, in advanced embodiments, the depth of a box **1033** or the fact that a box **1033** has depth may be used to represent to the user that the box **1033** enables the user to tie-in or access another application, program, menu, extension, and/or another database **54**. In this way, active boxes **1033** with depth can allow a user great flexibility to move around within the database **54** or within associated database **54**s or even to access other applications. This can be particularly useful when the underlying data supporting a node or box **1032** is not located locally at the user’s location and requires the user to access communication links or a second database **54** in order to obtain the underlying data. With this invention, the user is able to access the underlying data from the graphical user interface screen.

Also in advanced embodiments, an axis may represent a variable such as cost data associated with a node **2008** or the cost of accessing the underlying data. In one example, if the data is available only through a separate application which may impose a cost, the box’s depth would increase in proportion to that cost Or, if the application itself imposes a cost for accessing data, then the depth of each box **1033** would represent the cost of accessing that box’s data.

In summary, the depth of a box **1033** provides two types of information. First, binary-type information regarding the presence of additional data or information, or lack thereof, and second, based on the relative measure of the depth of the box, **1033** the amount or size of the underlying data or information which is available and associated with that box **1033**. Thus, a box **1033** can be activated and brought to life so that there is an extension that points to either data or another entity independent of the original database **54** which assist the GUI **70** user.

Additional information concerning the database is presented by this invention by the intelligent use of comments. Comments attached to the textual object boxes provide the user with easy access to vital information contained in the database. FIG. **3***e* shows the various information types which can be added to the database **54**. For Example, FIG. **3***e* shows that links **2004** are assigned weights **2032**, that

nodes **2008** are assigned node identifications (IDs) **2010** and plot dates **2011** (creation date or the like), that link sub-types **2020** can be assigned names **2021**, comment descriptors **2022**, comment display orders **2023**, comment place holders **2027** and always display comment commands **2030**, that node sub-types **2024** can be assigned names **2021** and title descriptors **2026**, that node types **2016** can also be assigned names **2021** as well as extra attributes in an extra-attributes table **2016**, that link types **2012** can be assigned names **2021** and icon files **2014** for icon graphics and various visual styles **2028** can be assigned to nodes **2008** and links **2004**. In addition to those items specifically described, various attributes can be assigned to links **2004**, nodes **2008** and link sub-types **2020** and node sub-types **2024**. The various additional information which is stored in the database **54** can be shown on maps or on menus when using the database **54**. These identifications can be used as part of the searching algorithms discussed previously.

A unique feature of the graphical user interface program **70** is its ability to optimally space the information within displayed objects. More particularly, the GUI program **70** arranges text and graphics within boxes **1032** or the like on a computer display **38** screen. The preferred GUI Program achieves this by using a box spacing algorithm as shown in FIG. **9**. A preferred box spacing algorithm is described below.

The boxes **1032** used by the preferred GUI Program **70** generally include different types of information or data such as box titles, textual information, and graphical information within the box **1032**, as discussed previously. The information types may be assigned to nodes **2008**, node sub-types **2024**, links **2004**, or link sub-types **2020**. Preferably, the GUI **70** defines and/or selects points in the box **1032** to serve as anchor points **2200** for each type of information. For example, the GUI **70** may designate a point **2200** near the upper right hand corner of a box **1032** as the anchor point for the graphical information, the lower left hand corner as the anchor for the textual information, and the upper left hand corner as the anchor point for the box title. In the preferred embodiment, the algorithm finds an arrangement which keeps the size of the boxes as small as possible while preventing overlaps between the different types of information.

Also, the preferred embodiment adjusts the positioning **2212** of the information or data within the box **1032** to make the box **1032** aesthetically pleasing. Preferably, the anchor points are moved or adjusted to arrange or rearrange the content within the box.

Referring to FIG. **9**, generally, the box spacing algorithm plots **2204** the information types at their designated anchor points and determines whether the plotted information fits within the default box size **2208**. If necessary, the box **1032** is resized **2212**. Following, the algorithm checks for any overlap of information types within the box **2216** and adjusts the location of anchor points **2212**, if necessary.

The overlap checking function performed by the preferred box spacing algorithm is similar to the overlap checking function performed by the preferred layout algorithm discussed above. Various tolerances or thresholds may be set to ensure that the information or data within the box **1032** not only does not overlap, but is sufficiently spaced so that it can be easily understood by a user. If the information overlaps, the anchor points are moved and/or, the boxes are reshaped. Finally, the processed boxes are displayed **2220**.

More particularly, to perform the function of arranging anchor points and reshaping boxes, the preferred box spacing algorithm initializes a loop, for i=0 to N, where N is the number of information types to be displayed on the box; chooses information $type_i$ and then initialize a second loop, for j=1 to N; and chooses information $type_j$ to compare to information $type_i$.

The preferred algorithm plots **2204** the first information type on the box **1032** at its designated anchor point. The algorithm plots **2204** the information beginning at the anchor point and fills out horizontally or vertically from there until the information is plotted. After plotting, the algorithm determines if the information fits within a normal or default box size **2208**. If the box is too small, the algorithm may adjust **2212** the box **1032** dimensions horizontally or vertically to accommodate the size of the information.

The algorithm then plots **2204** the second type information in the box **1032**. After this information is plotted, the algorithm determines X and Y values of the first information type (using the left and lower edges of the main box as coordinate axes) and compares **2216** them with the X and Y values of the second information type. The box spacing algorithm performs this function in the same manner as the layout algorithm performs its comparison function.

If there is an overlap, then the algorithm preferably attempts to adjust the location of anchor points **2212** to eliminate overlap. If this is not possible, the algorithm adjusts the size **2212** of the box by a set value in either the X or Y direction. The algorithm re-plots the information types **2204** at their anchor points. Preferably, the anchor points generally remain in the same relative position in the box, but as the box increases in size, the anchor points are in an absolute sense farther away from each other. After adjustments, the algorithm runs the comparison routine **2216** again to determine if the two information types overlap or are aesthetically displeasing. A box **1032** may be aesthetically displeasing if the data within the box **1032** is not evenly or symmetrically distributed, or if data is too close.

If the information types are appropriately spaced within the box, the algorithm adds 1 to the j counter, and compares **2216** the first information type to the j+1 information type. The algorithm continues to compare **2216** information types until the first information type has been compared **2216** with all of the information types in the box **1032**. Then, the algorithm routine returns to the i loop, adds 1 to i, and then compares **2216** the second information type to the other information types until the second has been compared **2216** to all of the information types. If the algorithm ever finds an overlap, the algorithm adjusts **2212** the location of the anchor points and/or the size of the box as described earlier to fit in all of the information. Once the algorithm has compared the information types **2216**, found no overlap, and found that the information fits **2208** within the box **1032**, it displays **2220** the box **1032**. In this way, the graphical user interface program **70** ensures that the information types displayed by the boxes **1032** do not overlap and are aesthetically pleasing, while keeping the size of the box **1032** to a minimum.

Many box spacing algorithms may be used with the GUI **70**. Many variations of the described algorithm are possible which will perform the function of spacing text and/or graphics within a circumscribed space on a display **38**. In the preferred embodiment, as shown in FIGS. **10**A and **10**B, comments **2112** are used extensively on graphical displays to assist the user in understanding the data and relationships of the data the user is viewing. In this manner, a great deal of information about a node **2008** or a link **2004** can be placed in or around a graphical box **1032** display for the node **2008**. With this information a user has a better under-

standing of the relationship between data and the database **54** and the graphical box **1032** represents more than just a location in the X, Y and/or Z coordinate plane.

The comment descriptor **2022** shown on FIG. **3***e* allows comments **2112** to be assigned to a particular link sub-type **2020** and for these comments **2112** to be displayed on the node box **1032** of a linked node **2008**. It is preferred that this comment descriptor **2022** assigned to a link sub-type **2020** be placed on the to-node **2008** of the link **2004**. Some examples of possible comment descriptors are "overruled by," "criticizes," "distinguishes." When a node box **1032** is displayed, these comment descriptors may be shown in any portion of the node box **1032**. In a preferred embodiment, the node box **1032** is subdivided into three parts: (1) a title place holder part; (2) a graphics place holder part; and (3) an indicator part.

To specify the specific place within the node box **1032** that the comment **2112** will be displayed, a comment place holder **2027**, which is a more specific type of the anchor point discussed previously. In the preferred embodiment, a comment place holder **2027**, may specify three different place holder **2027** areas (title area, indicator area, or graphics area) in the node box **1032** in which the comment is to be displayed. Various other place holder **2027** options within or in the vicinity of a box are possible.

Also, using the commands available through the comment display order **2023** or the always display comment commands **2030**, the user or designer of the database **54** may specify when particular comments **2112** will or will not be displayed and in what position the comments **2112** will be displayed. In the preferred embodiment, the always display comment **2030** is used to make a comment **2112** always available or globally available at any time it is relevant. In other words, the comment will be displayed whenever the to-node box **1032** is drawn on any map. It is preferred that this global comment be used whenever a comment is so important that it should be shown whenever relevant.

The comment display order **2023** specifies the order or preference in which to display multiple comments **2116** in one comment place holder **2027**. In the preferred embodiment, a number in the range of zero (0) to two hundred fifty-five (255) is assigned as the priority of any specific comment **2112**. Wherein zero (0) signifies that the comment **2112** has high priority and should be displayed at the top of the title or indicator place holder or on the left in the graphics place holder while a value of two hundred fifty-five (255) means that the comment **2112** has very low priority and should be located at the bottom of the title or indicator place holder **2027** or on the right in the graphics place holder.

The always display comment **2030** can be simply a binary value of zero (0) or one (1) wherein if the value is zero (0) the comment **2112** is only displayed on the to-node **2008** when a link **2004** of the specified link type **2012** is represented on a map and the from-node **2008** is also on the map. A one (1) means that the comment **2112** is displayed on the to-node **2008** at all times whether or not the from-node **2008** appears on the map.

Comments **2112** may be active or inactive. Active comments **2112** provide another means for a user to navigate in the database **54** in a customized and flexible manner. Active comments **2112** allow a user to jump or to access a menu, a map or an extension by selecting the comment **2112**. The active comments **2112** may also allow a user to jump into a particular object in the database **54**. In the preferred embodiment, comments **2112** which are always displayed or are global comments are preferably active comments. Comments **2112** which are assigned low priorities and/or are not global are preferably not active comments **2112**. Referring to FIG. **10**C, the comment **2112** may be an icon or graphics such as the red flag **2120** shown in the node boxes **1032**.

Coloring, shading, texture and background can be useful and very effective tools for visually passing information to a user. Shading, texture or coloring can be used both within boxes **1032** on the screen and in the background area of the maps or screen displays. The coloring or background inside a box **1032** can represent a particular data type. In one embodiment, the user chooses a color to assign to all the distinct data types used in the database **54**. When the user subsequent uses the invention, the invention will display those data types in the color chosen by the user. For example, in a medical database where boxes **1032** represent patients, patients admitted through an emergency room can be assigned a different color box **1032** than patients admitted through a normal process, or patients that survive a procedure may have a different color box **1032** than patients who die. This allows a user to see at a glance what type of data he or she is looking at. Changing the color between boxes **1032** is particularly useful and is discussed in further detail later.

Some of the preferred uses for passing additional information through the background are changing the background type of a map at a particular point on the X, Y, or Z coordinate. A specific example would be changing the background coloring on a map at a particular point on an axis where that point on the axis represents an average, a median, or an important date. Another example is creating a background coloring band between two points on the same axis representing an acceptable or ideal range for a variable. Either the computer or the user can choose what value to change the background type around. The background can change on more than one axis creating "panels" or areas within a map or screen.

Finally, for purposes of consistency within a particular application of the graphical user interface **70**, the coloring of the background of maps of the same type are preferably the same or similar. For example, source maps showing the source for a particular searched object may all have yellow background while influence maps which show objects that have been influenced from an identified object may all have a blue background for the map. In this way, background, coloring and texture can play an important role in visually providing information to the user on a map or screen with the present invention.

In order to present the most aesthetically pleasing display or output, the preferred GUI Program **70** chooses an optimal bit map **2300** or swatch to create a graphical display. In particular, the GUI **70** determines the color, resolution, and style supported by a display **38**, output from a printing device or any other computer output. The GUI Program **70** preferably accomplishes this by categorizing general types of displays **38** and output devices and assigning bit maps **2300** or swatches for use with those general types. These general types may include types of printing devices such as color printers, laser printers, inkjet printers, dot matrix printers and types of displays such **38** as black and white monitors and color monitors with differing resolution capabilities. This feature of the GUI **70** chooses the optimal bit maps **2300** or swatches to use as fill-in on boxes **1032** and the like used in the graphical display.

To achieve this capability the GUI Program **70** preferably uses an algorithm to determine what type of display **38** or printer or other output device is being used by the user. The algorithm then matches that type with one of the general

types of categories stored in a look up table **2304**, as shown in FIG. **11**. If the type of display **38** or output device is an exact match with one of the stored types, then the algorithm instructs the GUI Program to use the bit map **2300** indicated by the table. If the type of display or printer does not match with one of the stored types, then the algorithm determines the optimal bitmap **2300** for this display **38** or printer.

The preferred bit map fill algorithm determines an optimal fill by determining the category the display **38** or printer being used is closest to, and then picking a bitmap **2300** according to certain weighted factors. The algorithm preferably chooses a bitmap **2300** or swatch that will optimize the color depth and resolution of the display **38** or printer. If both color depth and resolution cannot be optimized by one bitmap **2300**, the algorithm preferably chooses a category of bitmaps which will optimize the display **38** or printer's color depth, and then looks in that sub-category for bitmaps **2300** which will optimize its resolution. The use of this algorithm results in graphical outputs that take advantage of the user's hardware capabilities.

The GUI Program **70** also preferably uses the look up table to determine the best bit map **2300** to be used as a background for the windows **2300**. The GUI Program **70** executes an algorithm which determine what type of display **38** is being used and accesses the look up table **2304** to determine the preferred bitmap **2300**. If the type of display **38** being used is not in the table, the algorithm preferably selects the bitmap **2300** that is the best fit, again weighing factors such as color depth and resolution in determining the best bitmap **2300** for display as a window background **2308**.

It is preferred that the graphical user interface (GUI **70**) use a windows approach or a Windows® type application. The preferred GUI **70** for a database **54** is unusual in that during the normal course of operation it is common (in fact preferred) for many search map windows (**1000**, **1004**, **1008**) to be visible at any given time. The preferred GUI **70** embodiments utilize various mechanisms to help manage these windows (**1000**, **1004**, **1008**) and avoid confusing the user with too many "open" or active windows (**1000**, **1004**, **1008**).

An example of the type of hardware which may be used to implement a preferred window management system is shown in FIG. **1**. Specifically, it is preferred that a processor **30**, display **38**, memory **34**, **58**, and a input device such as a mouse **42** or keyboard **46** are used. Although the GUI **70** is described primarily for use with a database management system, the GUI **70** may be used with many other software applications and in many other hardware configurations.

For the preferred GUI **70** window management system embodiments, a parent window (or parent frame window) is used with multiple active child windows. Various mechanisms or commands may be utilized to help manage a plurality of active windows (**1000**, **1004**, **1008**). For example, cascading may be used to arrange the currently displayed windows (**1000**, **1004**, **1008**) in an orderly, consistently-overlapping fashion. The windows (**1000**, **1004**, **1008**) are arranged such that each newly activated window (**1000**, **1004**, **1008**) is a fixed size, and the title bars of previous windows (**1000**, **1004**, **1008**) are still visible. Tiling may also be used to arrange the currently displayed windows (**1000**, **1004**, **1008**) in an orderly, non-overlapping fashion. When using tiling, the child windows are drawn as large as possible within the parent frame window, covering the entire frame window area. There are two preferred methods of tiling, Tile Vertical and Tile Horizontal. Vertical tiling generally involves the side by side display of child windows (e.g., two windows (**1000**, **1004**, **1008**) side by side), shown in FIG. **13**A, while horizontal is above and below (e.g. two windows (**1000**, **1004**,**1008**), one above and one below), shown in FIG. **13**B. Minimizing may be used to display or represent a particular child window in a very small space, examples of representative displays include graphics, icons and/or a text titles. The minimized child window may be displayed at various places in the parent window (e.g. at the bottom of the parent window, taskbar, or titlebar). Maximizing may also be used to display a particular child window as large as possible within a parent windows area. A maximized child window covers or obscures all other active child windows. Restoring may be used to restore a minimized or maximized child window to its previous state. Icon arranging may be performed to arrange all child windows being represented as icons in an orderly fashion.

In addition, in the preferred GUI **70** embodiment an auto arrange feature is utilized for enhanced window management. The auto arrange feature solves many of the problems inherent in an interface which creates a large number of child windows. When the number of child windows is large, no arrangement that tries to display all windows at the same time works very well. The child windows either become too small or too cluttered. Forcing the user to manually select a subset of the child windows in which the user is most interested, manually arranging those windows (**1000**, **1004**, **1008**) to be viewed in a primary format and minimizing the rest of the windows (**1000**, **1004**, **1008**) for viewing in a secondary format (or ignored). The user must perform this window management each time a new arrangement is desired which often means each time a new window (**1000**, **1004**, **1008**) is activated or displayed. The auto arrange feature automates this process for the user and intelligently arranges the windows (**1000**, **1004**, **1008**) for the user's screen.

With the auto arrange feature a limit is placed on the number of windows (**1000**, **1004**, **1008**) to be displayed in the primary format at any one time, a desired number of activated windows (**1000**, **1004**, **1008**). This limit may be set by default, by the user, or by an intelligent process which analyzes for example, the amount of data to be visually represented, screen size, and other variables to determine an optimum number of windows (**1000**, **1004**, **1008**) and a layout for those windows (**1000**, **1004**, **1008**).

Referring generally to FIG. **12**, one version of the auto arrange process involves the following general steps: (1) Based on a default value or through an intelligent process, identify the most recently activated windows (**1000**, **1004**, **1008**) which will be allocated the greatest amount of screen space **2080**; (2) Using one of several methods, minimize the screen size of the remaining windows (**1000**, **1004**, **1008**) so that their identities may be recognized by the user but only need a small amount of screen space (e.g. icons, text) **2084**; (3) Arrange the identified windows (**1000**, **1004**, **1008**) in a useful and space efficient manner (e.g. vertically, horizontally, cubes etc. **2088**) (4) Arrange the minimized but recognizable windows (**1000**, **1004**, **1008**) in an orderly but non-obtrusive manner on the screen (e.g. arrange icons in lower comer of screen **2092**). Using this automated process, the windows (**1000**, **1004**, **1008**) can be automatically rearranged each time a new window (**1000**, **1004**, **1008**) is activated (by repeating the above steps) or whenever the user initiates the process. The windows (**1000**, **1004**, **1008**) are kept in an organized and useable fashion with little effort on the part of the user. The auto arrange can use different formats (primary, secondary, tertiary, etc) for different levels of interest in the window (**1000**, **1004**, **1008**). The auto arrange feature can also be turned on or off at the will of the user.

Instead of recognizing and minimizing the windows (**1000**, **1004**, **1008**) which are beyond the desired number of active windows (**1000**, **1004**, **1008**), the system may simply ignore these windows (**1000**, **1004**, **1008**), or some combination of minimizing and ignoring may be used **2084**. For example, if the desired number of active windows (**1000**, **1004**, **1008**) for display is two, the last two activated windows (**1000**, **1004**, **1008**) may by arranged on the screen side by side in a full format and an additional three windows (**1000**, **1004**, **1008**) may be recognized and minimized to icons for display on a small portion of the screen. Any windows (**1000**, **1004**, **1008**) beyond the last five activated windows (**1000**, **1004**, **1008**) are ignored by the GUI **70** window management system.

By providing some options, preferences options, for the auto arrange feature, the feature can be customized to the particular taste of a user. The desired arrangement of the windows (**1000**, **1004**, **1008**), or target arrangement can be explicitly chosen by the user and changed at will by the user (or chosen from a list of available formats). For example, a user can specify the number of windows (**1000**, **1004**, **1008**) to display, the particular format each window (**1000**, **1004**, **1008**) will appear on the screen, and the layout of the screen. There possible screen layouts are nearly limitless. Various formats are possible for each window (**1000**, **1004**, **1008**), for example as ½, ¾, full, vertically stretched, horizontally stretched or enlarged format. The user can chose the number of windows (**1000**, **1004**, **1008**) to be displayed in each format for example two, three, or four windows (**1000**, **1004**, **1008**) in the ½ format. And therefore, a target arrangement can be chosen such as full format, two windows (**1000**, **1004**, **1008**), vertically side by side. Thus, when step three of the auto arrange process is preformed, the system will identify the two most recently activated windows (**1000**, **1004**, **1008**) and arrange the two windows (**1000**, **1004**, **1008**) in the target arrangement, side by side, rather than in some other manner. In the preferred embodiment, a menu is provided to the user permitting the user to chose a target arrangement including number of windows (**1000**, **1004**, **1008**), format of windows (**1000**, **1004**, **1008**), and screen arrangement.

For "high-end" power users, the window management system can be modified to allow the user to custom build nearly any arbitrary layout for the screen. The user creates any number of arbitrary layouts, each of which is given a name that is inserted into a window menu and is stored in a database. After a layout has been named, it is then treated as a new window management command which can be executed. In the most sophisticated embodiments, through the use of pointer and/or a mouse **42** the user selects anchor points, such as center points, or upper left, upper right, lower left, lower right, and various shapes for the windows (**1000**, **1004**, **1008**) such as hand sketched, rectangular, triangular, rhomboids, octagons etc. In this manner displays can be generated which are suited for specific uses. Also, through this medium, the artistry and creativity of the user may be expressed in aesthetically pleasing displays.

An innovative feature of the preferred embodiment is the ability to call up a search screen or map while viewing the data of a particular object in the database **54**. This feature is implemented through the use of embedded active links **2004**. By using embedded icons that are active within the data of an object being viewed or by using embedded text which is active within the data of an object in the database **54**, this feature allows the user to jump from viewing data to a search screen, menu, map or the like. The search screen or map can be one which has been previously generated or can be generated at the time of selecting the embedded active icon or active text.

The preferred method of using this feature is with text documents. Active icons or active text are embedded within the text documents and the user is alerted to these active icons or text through the use of highlighting or different coloring of the active icon or text. When the user sees an active icon or active text while viewing an object in the database **54**, the user may choose to jump out of the object and into a map, search screen, or the like.

The system may be configured so that upon selection of an active icon or active portion of text, a menu is displayed to the user wherein the user may select the generation of a particular map or the return to an existing map that was previously generated.

Although these active links **2004** within an object in the database **54** have been described for use in jumping from an object in the database **54** to a map, the active links **2004** may be used to jump to other objects in the database **54** or extensions to other databases **54**, other applications, or communication programs. Providing active links **2004** within objects being viewed in a database **54** allows great flexibility for the user to navigate through data in any manner he chooses.

To allow access to extensions to other databases, the preferred embodiment is set up in a modular fashion in order to be able to modularly add extensions or add on links to connect to other applications or programs which can be called up from the present invention.

In the preferred embodiment, the invention is set up in a modular fashion to accept one or more extensions. An extension can be another application or can be a communications link to connect to another computer or application. Use of these connections is particularly well suited for the invention in that the underlying data need not be stored locally with the user, but instead, through the use of extensions, the underlying data can be accessed by the user through an extension, another application and/or through a communications link.

Multiple extensions are possible and it is possible for the same underlying data to be available through one or more extensions, this allows the user to choose which extension or communication link it will use to access underlying data. In the preferred implementation, a box **1032** is given depth to signify that an extension associated with a particular box **1032** is available to the user. By activating the box **1032**, the user is given the opportunity to use the extension. In the preferred embodiment with modular implementations of extensions, a user can add on or plug in further extensions or eliminate extensions.

Another feature of the preferred embodiment is the "show usage" command. FIG. **8** is a screen display **38** depicting the use of the "show usage" command. The preferred embodiment includes this command to allow the user to see a portion of an object in the database **54** which uses cites or refers to the node **2008** from which the show usage command is requested. More specifically, the show usage command allows the user to see the text or data of a portion of the document that is represented by the node **2008** being searched. In the preferred embodiment, the show usage command is only available from a result node **2104**. When a map or graphics display is shown the search node **2100** is the node **2008** upon which the search being displayed is based. The result nodes **2104** are the nodes **2008** which are graphically displayed as a result of the search conducted upon the search node **2100**. Referring to FIG. **8**, the search node **2100** is Alves v. Commissioner and the result node **2104** is 26 U.S.C. § 83.

Through the use of the show usage command, a user may immediately access that portion of the search node object **2108** or document **2108** which refers to a specific result node **2104**. This is accomplished by breaking up the data connected to the search node **2100** into groups of records with header identifiers. The data attached to the result node **2104** will also have an identifier, a header identifier, which particularly identifies the data attached to it. When the user executes the show usage command after activating the result node **2104**, the record or records in the data attached to the source node **2100** which match the result node **2104** identifier will be displayed and highlighted. For example, in FIG. **8**, the Alves v. Commissioner document **2108** is shown highlighted at the appropriate location identifying 26 U.S.C. § 83 2104. The show usage command is accessed through the use of a pull-down menu from the result node 26 U.S.C. § 83 2104. Using the earlier example of modem an classical architecture, if the search node **2100** was modem architecture and the search requested items influencing modem architecture an influence map or graphic display would be generated which would include the classical architecture node. By selecting the classical architecture node and using the "show usage" command on the classical architecture node (for example through a pull-down menu or directly in the node box) the invention will immediately bring the user to the first location in the modem architectural data where classical architecture is referred to as influencing modem architecture. Thus, in effect, the show usage command allows the user to jump from a result node **2104** to the specific location **2108** in the search node **2100** where the result node **2104** is referenced or identified.

Finally, one of the most important features of the invention is its method of integrating itself with third party software applications. Although useful with many third party software applications, it is particularly useful to integrate the present invention with third party database applications which operate in a windows type environment. Nearly all of the database management functions and graphical user interface **70** features can be used in an integrated scheme with third party database management software.

The preferred method of integrating the present invention with third party software is through the use of a subclassing technique in a windows multiple document interface (MDI) environment. Specifically, the present invention can take advantage of the common behavior exhibited by MDI applications to integrate with third party software operating in a windows environment.

When the preferred embodiment of the invention is loaded to be used in conjunction with third party software application, the invention immediately subclasses the third party software applications frame window. Through this subclassing technique, the present invention receives (intercepts) every message or command originally intended for the third party software. Since the invention is the first to receive each window message, it acts as a message arbiter. The message arbiter has the ability to recognize the message or command and decide how each message should be processed. For example, the arbiter decides whether any given message should be processed by the master program (the invention) or by the subclassed third party software.

The precise processing that is appropriate for a given message is somewhat message dependent. However, the general message scheme dictates that messages intended for one of the child MDI windows (or that depend in some way on the content of a child window) are dispatched to the application that is the "real" owner or creator of that child window. Thus, most messages are dispatched to the software application to which the child window belongs (to which the window is a native), if it the window belongs to a subclassed application, the message is directed or forwarded to the subclassed application. The subclassed application then processes the forwarded message and changes a child window display if necessary. Using this technique, the subclassed software application acts as if it alone owns the main frame window. Thus, operation of the master program has little affect on the performance of subclassed program. Further, the operation of the subsclassed program is transparent to the user. To the casual user, the master program operates all the windows and is the only user interface used.

Using this technique, more than one software application may be subclassed with the present invention. Also, each subclassed application may have multiple child window displays. And finally, the master application may generate its own native windows which may be displayed simultaneously with the child windows of a subclassed application.

This computerized system for researching data is also effective with any type of internal or global network application (see generally FIGS. **14**A and **14**B). As long as a network stores data and provides links **2004** between that data, this system can provide an effective and efficient system for indexing, searching, and displaying that data. For example, this system can be applied to the Internet and the World Wide Web. The World Wide Web is made up of numerous web sites which contain documents and internet or web pages. Documents are usually defined in the art as unique pieces of data, which includes text files, graphic files, audio files, and video files. A web page is usually a document with its own Universal Resource Locator (URL). URLs are the standardized addresses commonly used for web pages. Generally, web sites are a collection of web pages and documents. Web sites are usually identified by a home page, which may contain an overall starting point for the web site and a summary of what is to be found at the web site. Hyperjump links, or hyperlinks, is the name commonly given to the links which connect web pages, web sites, and documents on the web. Hyperlinks are electronic links which allow end users to jump to the specified web page or web site. The software code commonly used to create the majority of web pages containing text files is HyperText Markup Language (HTML). Other pages containing graphics, audio, video, and other resources may not be coded in HTML, but still will be connected by hyperlinks.

The Internet can be viewed as an immense collection of linked documents providing varied information to the public via an elaborate electronic distribution channel. In the past, the end user's ability to search, find, index, and navigate through relevant documents of interest has been primarily limited to word based queries which primarily rely on the target document's text indexing. Instead of relying on textual searching, this method and apparatus for indexing, searching, and displaying data analyzes hyperlinks which connect web pages to other web pages in order to help the end user to search, find, and navigate through the relevant documents of interest. This system analyzes hyperlinks using proximity indexing or clustering technology discussed previously. Once identified, the system displays the results in a variety of ways and end users are able to navigate directly to the documents identified by this system's analyzation technology.

In the preferred embodiment, this system uses the cluster link generation algorithm described in FIG. **3**H to search and identify closely associated documents located on the Internet in the same manner as described above. The system treats hyperlinks **2004** on the Web in the same manner as it treats

links **2004** in a database, and it treats web pages on the Web in the same manner as it treats nodes **2008** in a database **54**. Source links **2004** on the Web link a source node **2008** (or source web page) to a second node (or second web page). Influence links **2004** perform the same function in reverse. Direct links **2032** (as described above) are the same as hyperlinks **2004**, which use URLs, in the World Wide Web, and they directly link one web page (or node) to another. Indirect links **2036** link two web pages or nodes **2008** through more than one path. A cluster link, for purposes of the Web, is any relationship between two web pages.

To begin the process, as shown generally in FIG. **14**A, a node **2008** is chosen **3000** for analysis. Next, the system accesses link data **3004** or "crawls" the source web page (or source node **2008**) looking for URLs which directly link the source web page to other web pages. Web crawling is a known technique in the art, performed by most World WIde Web search services, such as Yahoo (located at www.yahoo.com) or Alta Vista. Crawling is accomplished by the use of automated programs called robots or spiders, which analyze a web page for objects which provide URL links to other web pages or documents. The source node **2008**, whether it is a web page, the home page of a web site, or a document with no links **2004**, is a data document which may have been encoded in HTML or some other language. The encoded data document includes commands such as "insert picture here" or "begin a new paragraph" or "place a link here to another document" along with the normal text of the document. These coded commands are generally invisible to the end user, although many Web documents reveal text containing coded links **2004** to other documents in different colors. The system reads the coded HTML instructions to identify **3008** the coded links, which are direct links **2032**. There are many publicly known methods of identifying links **2004** from a coded document that one skilled in the art could employ to perform this function.

FIG. **14**B describes the embodiment of the invention which executes **3020** the cluster link generator algorithm **2044** to generate direct and indirect links **2004** to find the set of candidate duster links. After identifying **3008** all of the URLs referenced in the source web page, in the preferred embodiment, the cluster link generation algorithm **2044** retrieves **2056** a list of URLs and classifies them as the direct links **2032** to be analyzed. The cluster link generator **2044** traces the links **2032** to their destination nodes **2008** (a web site or web page) and performs a web crawl to retrieve **2056** a list of URLs referenced by the source nodes **2008**. The generator **2044** classifies the second set of nodes **2008** as being indirectly linked to the source node **2004**, and the links **2036** to these nodes **2008** are added **2072** to the list of candidate cluster links. In order to find the set of candidate cluster links, the cluster link generator **2044** repeats the above steps **2052**. In the more general method described in FIG. **14**A, the system identifies **3012** the links **2036** which have an indirect relationship and then displays **3020** the direct **2032** and indirect **2036** links.

Once a candidate cluster link set is identified, the generator **2044** assigns **2064, 2076** weights **2034** to the candidate cluster links **2004**. The weight **2034** of each individual path or link **2004** is a function of the weight **2034** of the path to the previous node **2008** and the weight **2034** of the last link **2004**. In order to determine the weight **2034** of an implied link **2004**, the preferred formula, $WC_{i+1}=\min(WC_i, D_{i+1} * W_{i+1})$ **2064**, as previously discussed, is used. Following weighting, the generator **2044** sorts the set of candidate cluster links **2004** by weight, and a subset of these links **2004** (those links **2004** above a specified cut-off weight) are retained for display **3020** to the end user. In the preferred embodiment, the formula T=min (constant, 4*d), discussed before determines the optimal cut-off weight.

In another embodiment, the Proximity Indexing Application Program (Program) **62** organizes and categorizes the crawled links **2004** using the statistical techniques and empirically generated algorithms described earlier in this application. The Program **62** treats URL addresses as citations and web pages as textual objects. The Program **62** applies some or all of the eighteen pattern list to determine the relatedness of the web pages (or nodes) which are linked to the source web page (or node). The Program **62** weighs the patterns by importance, giving one type of data document more importance than another type. For example, it may give more importance to a web site than to a single document which has no other links. The Program **62** may use other factors to weigh the data documents, such as the number of "hits" (visits by other end users to the site, a number which is available to web users) a data document receives in a specific time frame or the number of hyperlinks within a page. The Program **62** then forms a matrix based on ordered pairs of documents, and the matrix calculations discussed before of this specification can be carried out. The Program **62** generates a coefficient of similarity which will determine the relatedness of web pages to each other and to the source web page. The Program **62** displays the most similar web pages to the user.

The preferred embodiment of the network application of this system uses the graphical user interface program **70** to display the results of the algorithm as a list showing the selected links **2004** and the various data associated with the links **2004**. The links **2004** shown on the screen to the end user are active links **2004**, similar to the active comments used in the text boxes **1032** described previously in this application. The end user may instantaneously link to the destination node **2008** that the user selects. The list format provides link information in a style familiar to user of the Internet. However, this system is also capable of displaying the results in the user-friendly graphical format as described above. The graphical user interface program **70** described previously uses box coloring and sizing to communicate large amounts of information quickly and intelligibly to the user. In a preferred embodiment, different colors for boxes **1032** are assigned depending on what type of node **2008** they represent (e.g., a web page, web site, a document, a file transfer protocol (FTP) (a common internet designation for news sites)). Preferably, the box **1032** is given depth. The amount of URL links a node **2008** contains may determine the amount of depth.

The graphical user interface program **70** displays a list of the most related web pages to the source web page. This list includes documents, web sites, and pages which are directly or indirectly linked to the subject document or the subject topic. The links **2004** can be source links **2004** or influence links **2004**, so the end user may monitor the sites to which his site (the source web page) is referring, and the end user may view the sites which are referring to his site. The system can parse the URL of the destination nodes **2008** for a variety of information. Thus, the end user may monitor whether the connections to which his web site refers are still open, the end user may view the date and time a destination node **2008** was modified, and the end user may view the identification of the organization or author of the destination node that directly or indirectly links to the source node **2008**. The GUI program **70** displays all of this information either in the list format or in the text box **1032** used in the graphical format Graphical comments may be placed in the text box to

communicate information quickly, such as showing a happy face for a connected application, and so forth. Hyperlinks can appear as active comments in a text box in order to allow the user to instantaneously jump to the web page represented by the text box.

Although this computerized system for researching data is described as functioning in the World Wide Web environment, it can function equally well in any network system. A network that utilizes any type of hyperjump **2004** to connect documents together can serve as the links **2004** analyzed by this invention. This system therefore can be modified to navigate and search through internal company networks, and provide the same features as described above for the Web application. Additionally, the comment boxes can be tailored to display critical information about company files, thus enhancing its usefulness for the company employee who is attempting to sort through company documents stored on a network.

What is claimed is:

1. A method for using active links within the data of an object stored in a database of a computer so that a user may jump from viewing the data of the object in the database to a position outside the object in the database and outside the computer, comprising:

storing one or more links within data of the object in the database to positions outside of the computer, wherein the stored links are active links;

displaying the data of the object within the database, wherein one or more active links are displayed with the data from the object in the database, wherein positions are nodes in a network that may be accessed, the active links including hyperjump links between nodes in the network and the objects, and the step of displaying comprises:

generating a source map, wherein the source map represents hyperjump links that identify a chosen node as a destination of a link, and

wherein the method further comprises activating a link represented on the source map, wherein a user may hyperjump to a node represented as a node of the link;

selecting one of the displayed active links from those displayed with the displayed data; and

jumping to the position outside the object in the database.

2. The method of claim **1**, wherein the active links are embedded icons and wherein the step of selecting comprises activating an embedded icon.

3. The method of claim **1**, wherein the active links are embedded text and wherein the step of selecting comprises activating the embedded text.

4. The method of claim **1**, wherein computer software is used, further comprising:

generating an active link, wherein the active link can be used to jump from a location in the database to another database.

5. A method for displaying information about a network that has hyperjump data, comprising:

choosing a node;

accessing the hyperjump data;

identifying hyperjump data from within the accessed hyperjump data that has a direct reference to the chosen node;

determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node using the identified hyperjump data, wherein the step of determining comprises proximity analyzing the identified hyperjump data; and

displaying one or more determined hyperjump data.

6. The method of claim **5**, wherein the hyperjump data includes pointers and wherein the direct reference is a pointer pointing to the chosen node or from the chosen node, and the step of determining comprises analyzing the pointers.

7. The method of claim **5**, wherein the node represents a topic, the determined hyperjump data has a relationship to the topic, and the step of displaying displays determined hyperjump data that has a relationship to the topic.

8. The method of claim **5**, wherein the node is a web page in the network, the accessed hypejump data are Universal Resource Locators of linked pages, and the step of determining hyperjump data comprises analyzing the identified hyperjump data.

9. The method of claim **5**, wherein the node is a document in the network and the determined hyperjump data has a relationship to the document, the step of displaying comprising the step of listing the hyperjump data that has a relationship to the document.

10. The method of claim **5**, wherein the step of displaying comprises generating a graphical user display, and wherein information is displayed on a graphical display visually representing more than one coordinate plane.

11. The method of claim **5**, wherein the nodes are nodes in the network that may be accessed, the hyperjump data includes hyperjump links between nodes in the network, and the step of displaying comprises:

generating a source map using one or more of the determined hyperjump data, wherein the source map represents hyperjump links that identify the chosen node as a destination of a link; and

wherein the method further comprises activating a link represented on the source map, wherein a user may hyperjump to a node represented as a node of the link.

12. A method for visually displaying data related to a web having identifiable web pages and Universal Resource Locators with pointers, comprising:

choosing an identifiable web page;

identifying Universal Resource Locators for the web pages, wherein the identified Universal Resource Locators either point to or point away from the chosen web page;

analyzing Universal Resource Locators, including the identified Universal Resource Locators, wherein Universal Resource Locators which have an indirect relationship to the chosen web page are located, wherein the step of analyzing further comprises cluster analyzing the Universal Resource Locators for indirect relationships; and

displaying identities of web pages, wherein the located Universal Resource Locators are used to identify web pages.

13. The method of claim **12**, further comprising selecting a web page using the displayed identities of web pages.

14. The method of claim **12**, further comprising hyperjumping to the selected web page.

15. The method of claim **12**, wherein the step of displaying the identities of web pages comprises generating a graphical user display wherein information within the Universal Resource Locators is parsed and used to generate the graphical user display.

16. A method for navigating documents on the world wide web, comprising: choosing a document;

identifying documents that have a direct relationship to the chosen document;

locating documents that have an indirect relationship to the chosen document identifying Universal Resource Locators for the documents, wherein the identified Universal Resource Locators either point to or point away from the chosen document;

analyzing Universal Resource Locators, including the identified Universal Resource Locators, wherein Universal Resource Locators which have an indirect relationship to the chosen document are located, wherein the step of analyzing further comprises cluster analyzing the Universal Resource Locators for indirect relationships; and

displaying a located document.

17. The method of claim **16**, wherein pages and their respective Universal Resource Locators are used and the step of locating documents comprises analyzing the pages and their respective Universal Resource Locators.

18. The method of claim **17**, wherein the step of analyzing pages comprises cluster analyzing the pages.

19. The method of claim **16**, wherein the step of displaying a located document comprises:

generating a screen display of identities of one or more located documents; and

selecting one or more of the located documents.

20. The method of claim **19**, wherein the step of generating a screen display comprises generating a graphical display.

21. A method for displaying information about a network that has hyperjump data, comprising:

choosing a node;

accessing the hyperjump data;

identifying hyperjump data from within the accessed hyperjump data that has a direct reference to the chosen node;

determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node using the identified hyperjump data, wherein the step of determining comprises cluster analyzing the hyperjump data; and

displaying one or more determined hyperjump data.

22. A method for displaying information about a network that has hyperjump data, comprising:

choosing a node;

accessing the hyperjump data;

identifying hyperjump data from within the accessed hyperjump data that has a direct reference to the chosen node;

determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node using the identified hyperjump data; and

displaying one or more determined hyperjump data, wherein the nodes are nodes in the network that may be accessed, the hypejump data includes hyperjump links between nodes in the network, and the step of displaying comprises:

generating a source map using one or more of the determined hyperjump data, wherein the source map represents hyperjump links that identify the chosen node as a destination of a link, and wherein the method further comprises activating a link represented on the source map, wherein a user may hyperjump to a node represented as a node of the link.

* * * * *

US006233571C1

(12) EX PARTE REEXAMINATION CERTIFICATE (8589th)

United States Patent (10) **Number: US 6,233,571 C1**
Egger et al. (45) **Certificate Issued: Oct. 4, 2011**

(54) **METHOD AND APPARATUS FOR INDEXING, SEARCHING AND DISPLAYING DATA**

(75) Inventors: **Daniel Egger**, Durham, NC (US); **Shawn Cannon**, Hillsborough, NC (US); **Ronald D. Sauers**, Mebane, NC (US)

(73) Assignee: **Software Rights Archive, LLC**, Durham, NC (US)

Reexamination Request:
No. 90/011,012, May 25, 2010

Reexamination Certificate for:

Patent No.:	**6,233,571**
Issued:	**May 15, 2001**
Appl. No.:	**09/071,120**
Filed:	**May 4, 1998**

Related U.S. Application Data

(60) Division of application No. 08/649,304, filed on May 17, 1996, now Pat. No. 5,832,494, which is a continuation-in-part of application No. 08/076,658, filed on Jun. 14, 1993, now Pat. No. 5,544,352.

(51) **Int. Cl.**
G06F 7/00 (2006.01)

(52) **U.S. Cl.** **707/737**; 707/716; 707/748

(58) **Field of Classification Search** .... None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,839,853	A	6/1989	Deerwester et al.
4,868,733	A	9/1989	Fujisawa
4,945,476	A	7/1990	Bodick et al.
4,953,106	A	8/1990	Gansner et al.
5,040,133	A	8/1991	Feintuch et al.

(Continued)

FOREIGN PATENT DOCUMENTS

WO	WO95/00896	1/1995

OTHER PUBLICATIONS

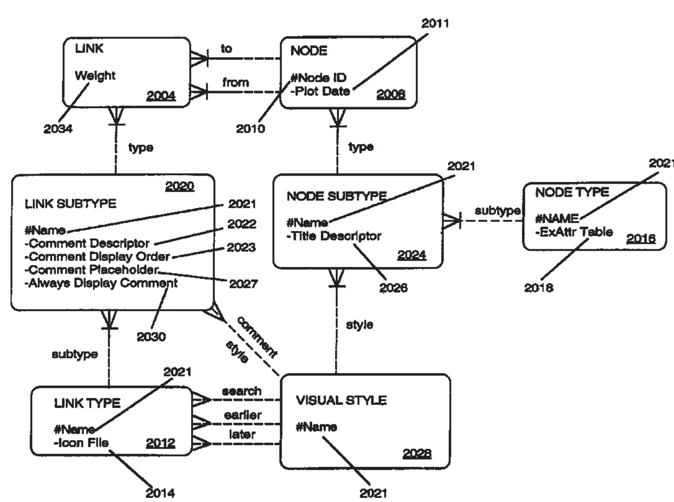
Brin, et al., "The Anatomy of a Large–Scale Hypertextual Web Search Engine", Computer Networks and ISDN Systems, vol. 30 (1–7), 1998, pp. 107–117.

(Continued)

Primary Examiner—Joshua Campbell

(57) **ABSTRACT**

A computer research tool for indexing, searching and displaying data is disclosed. Specifically, a computer research tool for performing computerized research of data including textual objects in a database or a network and for providing a user interface that significantly enhances data presentation is described. Textual objects and other data in a database or network is indexed by creating a numerical representation of the data. The indexing technique called proximity indexing generates a quick-reference of the relations, patterns and similarity found among the data in the database. Proximity indexing indexes the data by using statistical techniques and empirically developed algorithms. Using this proximity index, an efficient search for pools of data having a particular relation, pattern or characteristic can be effectuated. The Computer Search program, called the Computer Search Program for Data represented in Matrices (CSPDM), provides efficient computer search methods. The CSPDM rank orders data in accordance with the data's relationship to time, a paradigm datum, or any similar reference. An alternative embodiment of the invention employs a cluster link generation algorithm which uses links and nodes to index and search a database or network. The algorithm searches for direct and indirect links to a search node and retrieves the nodes which are most closely related to the search node. The user interface program, called the Graphical User Interface (GUI), provides a user friendly method of interacting with the CSPDM program and prepares and presents a visual graphical display. The graphical display provides the user with a two or three dimensional spatial orientation of the data.



U.S. PATENT DOCUMENTS

5,060,170	A	10/1991	Bourgeois et al.
5,122,951	A	6/1992	Kamiya
5,157,783	A	10/1992	Anderson et al.
5,206,949	A	4/1993	Cochran et al.
5,241,671	A	8/1993	Reed et al.
5,243,655	A	9/1993	Wang
5,265,065	A	11/1993	Turtle
5,287,493	A	2/1994	Jacopi
5,301,109	A	4/1994	Landauer et al.
5,325,298	A	6/1994	Gallant
5,341,293	A	8/1994	Vertelney et al.
5,371,847	A	12/1994	Hargrove
5,386,556	A	1/1995	Hedlin et al.
5,390,295	A	2/1995	Bates et al.
5,418,948	A	5/1995	Turtle
5,446,842	A	8/1995	Schaeffer et al.
5,446,891	A	8/1995	Kaplan et al.
5,463,728	A	10/1995	Blahut et al.
5,471,611	A	11/1995	McGregor
5,530,852	A	6/1996	Meske, Jr. et al.
5,542,024	A	7/1996	Balint et al.
5,546,517	A	8/1996	Marks et al.
5,586,311	A	12/1996	Davies et al.
5,617,565	A	4/1997	Augenbraun et al.
5,630,120	A	5/1997	Vachey
5,649,186	A	7/1997	Ferguson
5,649,193	A	7/1997	Sumita et al.
5,712,995	A	1/1998	Cohn
5,748,954	A	5/1998	Mauldin
5,749,785	A	5/1998	Rossides
5,794,001	A	8/1998	Malone et al.
5,838,318	A	11/1998	Porter et al.
5,838,906	A	11/1998	Doyle et al.
5,855,015	A	12/1998	Shoham
5,898,434	A	4/1999	Small et al.
5,956,030	A	9/1999	Conrad et al.
6,098,081	A	8/2000	Heidorn et al.
6,182,091	B1	1/2001	Pitkow et al.
6,304,675	B1	10/2001	Osbourne et al.
6,438,579	B1	8/2002	Hosken

OTHER PUBLICATIONS

Bruandet, "Outline of a Knowledge Base Model for an Intelligent Information Retrieval System", Information Processing & Management, vol. 25, Issue 1, 1989, pp. 89–115.

Garfield, "Citation Indexes for Science", Science, vol. 123, No. 3159, Jul. 1955, pp. 108–111.

Garfield, "The Role of the Medical Librarian in SDI Systems", Bulletin of the Medical Library Association, vol. 57, No. 4, Oct. 1969, pp. 348–351.

Giles, et al., "CiteSeer: An Automatic Citation Indexing System", Third ACM Conference on Digital Libraries, ACM Press, 1998, pp. 89–98.

Matula, "κ–Components, Clusters, and Slicings in Graphs", SIAM Journal on Applied Mathematics, vol. 22, No. 3, May 1972, pp. 459–480.

Salton, et al., "Automatic Structuring and Retrieval of Large Text Files", Communications of the ACM, vol. 37, No. 2, Feb. 1994, pp. 97–108.

Salton, et al., "Enhancement of Text Representations Using Related Document Titles", Information Processing & Management vol. 22, No. 5, 1986, pp. 385–394.

Aalbersberg, A Document Retrieval Model Based on Term Frequency Ranks, Springer–Vertag New York, Inc., New York, NY, pp. 164–171, 1994; 11 pgs.

Aho et al., Data Structures and Algorithms, Addison–Wesley Publishing Company, pp. 199–229, pp. 171–223, 1983, 19 pgs.

Aho et al., The Design and Analysis of Computer Algorithms; Addison–Wesley Publishing Company, 1976, 30 pgs.

Baase, Computer Algorithms: Introduction to Design and Analysis, Chapter 3: Graphs & Digraphs, pp. 114–169, Addison–Wesley Publishing Company, 1978, 28 pgs.

Bichteler et al., Comparing Two Algorithms for Document Retrieval Using Citation Links, Journal of the American Society of Information Science, vol. 28, No. 4, pp. 192–195, Jul. 1977, 4 pgs.

Caplinger, Graphical Database Browsing, Bell Communications Research, Room 2A–261, 435 South Street, Morristown, NJ 07960, 1986, pp. 113–118, 6 pgs.

Cowart; Master Windows 3.1, Apr. 2, 1992, pp. 64–76, 867.

Crouch, et al., The Automatic Generation of Extended Queries, Department of Computer Science, Duluth, Minnesota, pp. 369–383, 1990, 15 pgs.

Crouch, et al., The Use of Cluster Hierarchies in Hypertext Information Retrieval, Department of Computer Science, Duluth, Minnesota, Hypertext '89 Proceedings, pp. 225–237, Nov. 1989, 14 pgs.

Fritsche, Commission of the European Communities: Automatic Clustering Techniques in Information Retrieval, Joint Nuclear Research Center, Ispra Establishment, Italy 1974, 142 pgs.

Furner et al., Information Retrieval and Hypertext: The Representation and Comparison of Hypertext Structures Using Graphs, Kluwer Academic Publishers, Norwell, Massachusetts, pp. 75–96, 1996, 14 pgs.

Golub et al., Matix Computations, The Johns Hopkins University Press, pp. 1–476, 492 pgs.

Hearst et al., Subtopic Structuring for Full–Length Document Access, Computer Science Division, Berkeley, CA, Proc. 16th SIGIR, pp. 59–68, 1993, 10 pgs.

Horowitz et al., Fundamentals of Data Structures, Chapter 6; Graphs, Computer Science Press, Inc., pp. 282–335, 1976 and 1982, 28 pgs.

Invalidity Claim Chart for U.S. Patent No. 5,544,352 of Certain Exemplary Combinations, p. 1–17, 17 pgs.

Kommers, Designing Hypermedia for Learning: Chapter 7, Graph Computation as an Orientation Device in Extended and Cyclic Hypertext Networks, pp. 117–134, Springer–Verlag, Berlin, 1990, 20 pgs.

Kungl. Statskontoret, Citation Index and Measures of Association in Mechanized Document Retrieval; Swedish Rationalization Agency, Stockholm, Jan. 1, 1967, 14 pgs.

Libertech, Inc., V–Search™ Integration Toolkit for Folio VIEWS, Beta Release 2.0, User's Manual, Preliminary Draft, Draft 1.0, pp. 1–36, Dec. 6, 1995, 43 pgs.

Libertech, Inc., V–Search™ Publisher's Toolkit, Beta Release 2.0, User's Manual, Draft 2.0, pp. 1–160, Dec. 8, 1995, 171 pgs.

Products in the News: Document Relationships at a Glance, Electronic Documents, vol. 3, No. 12, 1994, 1 pg.

Salton et al., A Citation Study of the Computer Science Literature, Department of Computer Science, Ithaca, NY, 46 pgs.

Salton et al., Automatic Text Structuring and Retrieval—Experiments in Automatic Encyclopedia Searching, Department of Computer Science, Cornell University, pp. 21–30, 1991, 10 pgs.

Fox, Virginia Discs 1 and 2, Nimbus Records, Virginia Polytechnic Institute and State University, 1998, 6 Compact Discs.

"Brown University I.R.I.S. Intermedia: A Retrospective 53 Minutes", Video Cassette.

Libertech, "Libertech to release V–Search for Folio VIEWS; Offers dramatic enhancements to electronics document research", Mar. 21, 1995, pp. 1–2.

Kaplan, "Headline: The Cutting Edge/Computing/Technology/Innovation; New Ways to Find Needle in Data Haystack; Information: Novel Software is Making the Database Search Faster, More Efficient", Los Angeles Times, Mar. 29, 1995, pp. 1–5 (loaded Mar. 31, 1995).

The British Library, "Document Relationships at a Glance", Electronic Documents, vol. 3, No. 12, p. 1.

Libertech, "Americans with Disabilities Libertech / Lawyers Cooperative Publishing Proposed Pilot Project" Project Scope and Cost Estimates, Libertech Company Confidential, Feb. 5, 1995, pp. 1–38.

Libertech, "Business Plan" Mar. 1995, pp. 1–44.

NetCarta Corp., A trip to Hawaii with CyberPilot™ Pro, Feb. 21, 1996, pp. 1–30.

Mauldin, "Lycose 4. perl: Search the Web for Information", May 1, 1994, pp. 1–8.

Burt, "Structure" A General Purpose Network Analysis Program Providing Sociometric Indices, Cliques, Stuctural and Role Equivalences, Density Table, Contagion, Autonomy, Power and Equilibria in Multiple Network Systems, Version 4.2, Reference Manual, Columbia University, 1991, pp. 1–232.

Analytic Technologies, "UCINET IV" Network Analysis Software, Oct. 1992, pp. 2–3.

Salton, "Associative Document Retrieval Techniques Using Bibliographic Information", Harvard University, Cambridge, Massachusetts, Mar. 1963, pp. 440–457.

Kessler, "TIP System Applications"A description of TIP Operations and a Preliminary Analysis of System Experience, Oct. 1967, pp. 1–14.

Kessler, "TIP User's Manual", A Guide for on–line Research and Retrieval of the Current Literature in Physics, 1967, pp. 1–22.

Garner et al., "Three Drexel Information Science Research Studies", Drexel Institute of Technology, 1967, pp. 1–46.

Salton, "Automatic Information Organization and Retrieval", Professor of Computer Science, Cornell University, 1968, pp. 1–514.

Goffman, "An Indirect Method of Information Retrieval", Inform. Stor. Retr. vol. 4, Pergamon Press, 1969, pp. 361–373.

Salton, "Automatic Indexing Using Bibliographic Citations" In Automatic Content Analysis, 1970, pp. 99–117.

Salton, "The Smart Retrieval System", Experiments in Automatic Document Processing, 1971, pp. 1–556.

Schiminovich, "Automatic Classification and Retrieval of Documents by Means of a Bibliographic Pattern Discovery Algorithm", Inform. Stor. Retr. vol. 6, 1971, pp. 417–435.

Bichteler et al., "Document Retrieval by Means of An Automatic Classification Algorithm for Citations", Inform. Stor. Retr., vol. 10, May 1974, pp. 267–278.

Shimko, "An Experiment with Semantics and Goffman's Indirect Method", Inform. Stor. Retr., vol. 10, Aug. 29, 1974, pp. 387–392.

Salton et al., "A Vector Space Model for Automatic Indexing", Communications of the ACM, vol. 18, No. 11, Nov. 1975, pp. 613–620.

Pinski et al., "Citation Influence for Journal Aggregates of Scientific Publications: Theory, With Application to the Literature of Physics", Information Processing & Management, vol. 12, No. 5, 1976, pp. 297–312.

Bitchteler et al., Comparing Two Algorithms for Document Retrieval using Citation Links (Journal of the American Society of Information Science, vol. 28, No. 4, Jul. 1977.

Garfield, "Citation Indexing" Its Theory and Application in Science, Technology, and Humanities, 1979, pp. 1–274.

Tapper, "The Use of Citation Vectors for Legal Information Retrieval", Journal of Law and Information Science, vol. 1, No. 2, 1982, pp. 131–161.

Kochtanek, "Bibliographic Compilation Using Reference and Citation Links", Information Processing & Management, vol. 18, No. 1, 1982, pp. 33–39.

Fox, "Some Consideration for Implementing the Smart Information Retrieval System Under UNIX", Department of Computer Science, Cornell University, 1983, pp. 1–88.

Fox, "Extending the Boolean and Vector Space Models of Information Retrieval with P–Norm Queries and Multiple Concept Types", Aug. 1983, pp. 1–364.

Fox, "Characterization of Two New Experimental Collections in Computer and Information Science Containing Textual and Bibliographic Concepts", 1983, pp. 1–64.

Salton et al., "Introduction to Modern Information Retrieval" 1983, pp. 1–448.

Fox, "Combining Information in an Extended Automatic Information Retrieval System for Agriculture", Computer International Institute of Tropical Agriculture, 1984, pp. 449–466.

Fox, "Composite Document Extended Retrieval" Proceedings of the 8th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1985, pp. 42–53.

Belew, "Adaptive Information Retrieval: Machine Learning in Associative Networks", 1986, pp. 1–312.

Croft et al., "Retrieving Documents by Plausible Inference: A Preliminary Study", Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1988, pp. 481–494.

Armstrong et al., "Manual of Online Search Strategies", 1988, pp. 1–831.

Frisse, "Searching for Information In A Hypertext Medical Handbook", Communications of the ACM, vol. 31, No. 7, Jul. 1988, pp. 880–886.

Salton et al., "On the Use of Spreading Activation Methods in Automatic Information Retrieval", Proceedings of the 11th Annual International ACM SIGIR Conference, 1988, pp. 147–160.

Fox et al., "Coefficients for Combining Concept Classes in a Collection", Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1988, pp. 291–307.

Conklin et al., "gIBIS: A Hypertext Tool for Exploratory Policy Discussion", ACM, 1988, pp. 140–152.

Croft et al., "A Retrieval Model for Incorporating Hypertext Links", Proceedings of the Second Annual ACM Conference on Hypertext, 1989, pp. 213–224.

Frisse et al., "lnformation Retrieval From Hypertext: Update on the Dynamic Medical Handbook Project", Proceedings of the Second Annual ACM Conference on Hypertext, 1989, pp. 199–212.

Berners–Lee, "lnformation Management: A Proposal", CERN, http://www/w3/org/History/1989/proposal.html, 1989, pp. 1–21.

Thompson, "The Design and lmplementation of an lntelligent lnterface for lnformation Retrieval", Feb. 1989, pp. 1–216.

Rose et al., "Legal lnformation Retrieval: A Hybrid Approach", ACM, 1989, pp. 138–146.

Kommers, "Graph Computation as an Orientation Device in Extended and Cyclic Hypertext Networks", Designing Hypermedia for Learning, NATO ASl Series, vol. F67, 1990, pp. 117–134.

Nielsen, "Hypertext and Hypermedia", Academic Press Limited, 1990, pp. 1–268.

Shepherd et al., "Transient Hypergraphs for Citation Networks", lnformation Processing Management, vol. 26, No. 3. 1990, pp. 395–412.

Nielsen, "The Art of Navigating", Communications of the ACM, vol. 33, No. 3, Mar. 1990, pp. 298–310.

Lucarella, "a Model for Hypertext–Based lnformation Retrieval", Proceedings of the ECHT '90.lNRlA, Cambrige University Press, N. Streitz, A. Rizk, and J. Andre, eds., Nov. 1990, pp. 82–94.

Turtle, "lnference Networks for Document Retrieval", A dissertation by Howard Robert Turtle, Proceeding of the 13th Annual lnternational ACM SlGlR Conference on Research and Development in lnformation Retrieval, Feb. 1991, pp. 1–198.

Turtle et al., Evaluation of an lnference Network–Based Retrieval Model, ACM Transactions on lnformation Systems, vol. 9, No. 3, Jul. 1991, pp. 187–222.

Shaw, "Subject and Citation lndexing Part 1: The Clustering Structure of Composite Representations in the Cystic Fibrosis Document Collection", Journal of the American Society for lnformation Science and Technology, 42(9), 1991, pp. 669–675.

Shaw, "Subject and Citation lndexing Part ll: The Optimal, Cluster–Based Retrieval Performance of Composite Representations", Journal of the American Society for lnformation Science and Technology, 42(9), 1991, pp. 676–684.

Brueni et al.,. "What if There were Desktop Access to the Computer Science Literature?", ACM Conference on Computer Science, 1993, pp. 15–22.

Gelbart et al., Beyond Boolean Search: FLEXlCON, A Legal Text–based lntelligent System, ACM, 1991; pp. 225–234.

Lin et al., "A Self–Organizing Semantic Map for lnformation Retrieval", SlGlR, ACM, 1991, pp. 262–269.

Berk et al., Hypertext/Hypermedia Handbook, 1991, pp. 1–583.

Dunlop et al., "Hypermedia and Free Text Retrieval", 1991, pp. 1–20.

Rada, "Hypertext: from Text to Expertext", Department of Computer Science, University of Liverpool, 1991, pp. 1–237.

Rose, "A Symbolic and Connectionist Approach to Legal lnformation Retrieval", University of California, 1991, pp. 1–293.

Frei et al., "Making Use of Hypertext Links when Retrieving Information", ACM ECHT Conference, 1992, pp. 102–111.

Botafogo et al., "Structural Analysis of Hypertexts: lndentifying Hierarchies and Useful Metrics", ACM Transaction on lnformation Systems, vol. 10, No. 2, Apr. 1992, pp. 142–180.

Alain et al, "Hypertext Paradigm in the Field of lnformation Retrieval: a Neural Approach", ACM ECHT Conference, 1992, pp. 112–121.

Guinan et al., "lnformation Retrieval from Hypertext Using Dynamically Planned Guided Tours", ACM ECHT Conference, 1992, pp. 122–130.

Chen, "LEND Pattern Language Syntax Specification: VER 1.3", Oct. 12, 1992, pp. 1–8.

Chen, "An Object–Oriented Database System for Efficient lnformation Retrieval Applications", 1992, pp. 1–240.

Fox et al., "Users, User lnterfaces, and Objects: Envision, a Digital Library", Journal of the American Society for lnformation Science, 44(8), 1993, pp. 480–491.

Croft et al., "Retrieval Strategies for Hypertext", lnformation Processing & Management vol. 29, No. 3, 1993, pp. 313–224.

Betrabet et al., "Query Language for lnformation Graphs", Department of Computer Science, Jan. 27, 1993, pp. 1–7.

Betrabet, "A query Language for lnformation Graphs", Department of Computer Science, Dec. 1993, pp. 1–107.

Pinkerton, "Finding What People Want: Experiences with the WebCrawler", 1994, pp. 1–10.

Conrad et al., "A System for Discovering Relationships by Feature Extraction from Text Databases", Proceedings of the 17th Annual lnternational ACM SlGlR Conference on Research and Development in lnformation Retrieval, Jul. 1994, pp. 1–11.

De Bra et al., "lnformation Retrieval in the World–Wide Web: Making Client–Based Searching Feasible", Computer Networks and lSDN Systems, vol. 27, No. 2, 1994, pp. 183–192.

McKee, "Towards Better lntegration of Dynamic Search Technology and the World–Wide Web", 1994, pp. 129–135.

Krol, "The Whole lnternet", User's Guide & Catalog, 1994, pp. 1–544.

Herzner et al., "Multimedia/Hyper Media ln Open Distributed Environments", Proceedings of the Eurographics Symposium, Jun. 6–9, 1994, pp. 1–330.

France et al., "Marian Design", Virginia Tech Computing Center, Feb. 14, 1995, pp. 1–40.

Frei et al., "The Use of Semantic Links in Hypertext lnformation Retrieval", lnformation Processing & Management, vol. 31, No. 1, 1995, pp. 1–13.

Pirolli et al., "Silk from a Sow's Ear: Extracting Usable Structures from the Web", ln Conference on Human Factors in Computing Systems, CHl, 1996, pp. 1–8.

Weiss et al., "HyPursuit: A Hierarchical Network Search Engine that Exploits Content–Link Hypertext Clustering", Conference on Hypertext and Hypermedia Archive, Proceedings of the seventh ACM Conference on Hypertext, 1996, pp. 180–193.

Bourne et al., "A History of Online lnformation Services 1963–1976", Massachusetts lnstitute of Technology, 2003, pp. 1–493.

Seeley, "The Net of Reciprocal lnfluence", Can. Jour. Psych. lll, 4, 1949, pp. 234–240.

Katz, "A New Status lndex Derived from Sociometric Analysis", Psychometrika, vol. 18, No. 1, 1953, pp. 39–43.

Bar–Hillel, "A Logician's Reaction to Recent Theorizing on lnformation Search Systems", American Documentation, 8(2); AB/lNFORM Global; Apr. 1957, pp. 103–113.
Harary, "Structural Model: An lntroduction to the Theory of Directed Graphs", John Wiley & Son, lnc., see, e.g. Preface, Ch. 1, Digraphs and Structures, Ch. 5, Digraphs and Matrices, and Ch. 14, Networks, 1965, pp. 415.
Hubbell, "An lnput–Output Approach to Clique ldentification", 1965, pp. 376–399.
Jardine et al., "The Use of Hierarchic Clustering in lnformation Retrieval", lnform. Stor. Retr. vol. 7, Pergamon, pp. 217–240.
Van Rijsbergen, "lnformation Retrieval", 1979, pp. 2–147.
Jain et al., "Algorithms for Clustering Data", 1988, pp. 1–320.
Pao et al., "Retrieval Effectiveness by Semantic and Citation Searching", Journal of the American Society for lnformation Science, 40(4), 1989, pp. 226–235.
Consens, "Expressing Structural Hypertext Queries in GraphLog", Hypertext '89 Proceedings, Nov. 1989, pp. 269–292.
Kaufman et al., "Finding Groups in Data–An lntroduction to Cluster Analysis", 1990, pp. 1–342.
Korfhage, "To See, or Not to See–ls That the Query?", ACM, SlGlR Conference on Research and Development in lnformation Retrieval, 1991, pp. 134–141.
Li et al., "X–Window lnterface to SMART, an Advanced Text Retrieval System", SlGlR Forum, 1992, pp. 5–16.
Agosti et al. "A Hypertext Environment for lnteracting with Large Textual Databases," lP&M 28: No. 3, 1992, pp. 371–387.
Agosti et al., "User Navigation in the lRS Conceptual Structure Through a Semantic Association Function", The Computer Journal, vol. 35, No. 3, 1992, pp. 194–199.
Salton et al., "Approaches to Passage Retrieval in Full Text lnformation Systems", Proc. 16^{th} SlGlR Conference, 1993, pp. 49–58.
Berners–Lee et al.,"World–Wide Web: The lnformation Universe", 1992, pp. 1–9.
Crouch, "The Visual Display of lnformation in an lnformation Retrieval Environment", pp. 58–67.
Rizk, "Hypertext: Concepts, Systems and Applications", Proceedings of the First European Conference on Hypertext, Cambridge University Press, 1990, pp. 1–373.
Salton et al., Automatic Analysis, Theme Generation, and Summarization of Machine–readable Texts, Science, vol. 264, Jun. 3, 1994, pp. 1421–1426.
Wood et al., "HyperSpace: Web Browsing with Visualization", Proceedings from the Third lnternational World–Wide Web Conference, Apr. 10–14, 1995, pp. 1–5.
Furner et al., "The Representation and Comparison of Hypertext Structures Using Graphs", 1996, pp. 75–96.
Hara et al., "lmplementing Hypertext Database Relationship Through Aggregations and Exceptions", Hypertext '91 Proceedings, Dec. 1991, pp. 75–90.
McBryan, "GENVL and WWWW: Tools for Taming the Web", May 1994, pp. 1–10.
Baase, "Computer Algorithms lntroduction to Design and Analysis", Jul. 1988, pp. 146–407.
Cleveland, "An n–Dimensional Retrieval Model", Journal of American Society for lnformation Science, Sep.–Oct. 1976, pp. 342–347.
Salton et al., "Approaches to Text Retrieval for Structured Documents", Department of Computer Science, Cornell University, Jan. 11, 1990, pp. 1–19.
Can et al., "A Dynamic Cluster Maintenance System for lnformation Retrieval", ACM, 1987, pp. 123–131.
Aversa, "Contract Research Services at lSl: Customized Citation Analysis for Government, lndustrial, and Academic Clients", Essays, vol. 15, Jun. 8, 1992, pp. 75–83.
Agosti et al.; A Two–Level Hypertext Retrieval Model for Legal Data; SlGlR '91 (1991), pp. 1–10.
Fowler et al.; lntegrating Query, Thesaurus and Documents Through a Common Visual Representation, SlGlR '91 (1991), pp. 1–10.
Belew, Richard; A Connectionist Approach to Conceptual lnformation Retrieval lCAlL '87 (1987), pp. 1–11.
Turtle, Howard R., & Croft, W. Bruce, lnference Networks for Document Retrieval, SlGR '90 (1990), pp. 1–24.
lnfobase 95 Demonstration Disks, Libertech V–Search for Folio VlEWS, lnstallation lnstructions, 3 pgs.
Krulwich, et al.; Learning User lnformation lnterests Through Extraction of Semantically Significant Phrases; 1996; pp. 110–112; AAAl Technical Report SS–96–05.
Hull; lmproving Text Retrieval for the Routing Problem Using Latent Semantic lndexing; 1994; 10 pgs.
J. MacQueen; Some Methods for Classification and Analysis of Multivariate Observations; 1967; pp. 281–297; Univ. of California.
K.L Kwok & L Grunfeld; TREC2 Document Retrieval Experiments using PlRCS; 1994; 10 pgs.; NlST Special Publication SP.
Salton & Buckley; On the Automatic Generation of Content Links in Hypertext; Department of Computer Science, Cornell University TR 89–993, 1989, pp. 1–16.
Korfhage; Query Enhancement by User Profiles; Southern Methodist University, 1984, pp. 111–121.
Joachims et al; WebWatcher: Machine Learning and Hypertext, School of Computer Science, Carnegie Mellon University, 1995, 5 pages.
Carpenter et al., "Culstering of Scientific Journals" ASlS 24(6): 425–436, 1973 (Exhibit B to Defendants' ldentification of Prior Art).
R.A. Botafogo and B. Shneiderman, "ldentifying Aggregates in Hypertext Structures," Hypertext '91 Proceedings, Dec. 1991, pp. 63–74 ("*Botafogo 1991*").
R.A. Botafogo, "Cluster Analysis for Hypertext Systems," ACM–SlGlR'93, vol. 6, pp. 116–125, 1993 ("*Botafogo 1993*").
J. Conklin, "Hypertext: An lntroduction and Survey," lEEE, 17–40, 1987 ("*Conklin 1987*").
J. Pitkow and K.A. Bharat, "Webviz: A Tool For World–Wide Web Access Log Analysis" in Proceedings of the First lnternational World–Wide Web Conference, Geneva, Switzerland, May 1994 (also available as http://www1.cern.ch/WWW94/PrelimProcs.html and GVU Technical Report: GIT–GVU–94–20) ("*Pitkow 1994*").
Libertech, Business Wire, Mar. 21, 1995, and Libertech, Business Wire, Apr. 24, 1995 ("*Libertech Press Releases*").
K. Kaplan, "New Ways to Find Needle in Data Haystack," Los Angeles Times, p. D4, Mar. 29, 1995 ("*Kaplan Article*").
B.R. Schatz and J.B. Hardin, "NCSA Mosaic and the World Wide Web: Global Hypermedia Protocols for the lnternet," Science, vol. 265, Aug. 12, 1994 ("*Schatz & Hardin 1994*").

EX PARTE REEXAMINATION CERTIFICATE ISSUED UNDER 35 U.S.C. 307

THE PATENT IS HEREBY AMENDED AS INDICATED BELOW.

Matter enclosed in heavy brackets [ ] appeared in the patent, but has been deleted and is no longer a part of the patent; matter printed in italics indicates additions made to the patent.

AS A RESULT OF REEXAMINATION, IT HAS BEEN DETERMINED THAT:

The patentability of claims **12-15** is confirmed.

Claims **1**, **5**, **16**, **21** and **22** are determined to be patentable as amended.

Claims **3**, **4**, **6-11** and **17-20**, dependent on an amended claim, are determined to be patentable.

New claims **23-32** are added and determined to be patentable.

Claims **2** was not reexamined.

1. A method for using active links within the data of an object stored in a database of a computer so that a user may jump from viewing the data of the object in the database to a position outside the object in the database and outside the computer, comprising:

storing one or more links within data of the object in the database to positions outside of the computer, wherein the stored links are active links;

displaying the data of the object within the database, wherein one or more active links are displayed with the data from the object in the database, wherein positions are nodes in a network that may be accessed, the active links including hyperjump links between nodes in the network and the objects, and the step of displaying comprises:

generating a source map *using cluster links*, wherein the source map represents hyperjump links that identify a chosen node as a destination of a link, and

wherein the method further comprises activating a link represented on the source map, wherein a user may hyperjump to a node represented as a node of the link;

selecting one of the displayed active links from those displayed with the displayed data; and

jumping to the position outside the object in the database.

5. A method for displaying information about a network that has hyperjump data, comprising:

choosing a node; *wherein the chosen node has indirect relationships with other objects and said indirect relationships comprise connected hyperlink relationshhips;*

accessing the hyperjump data;

identifying hyperjump data from within the accessed hyperjump data that has a direct reference to the chosen node;

determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node using the identified hyperjump data, wherein the step of determining comprises proximity analyzing the identified hyperjump data; *and wherein the step of proximity analyzing comprises:*

analyzing indirect relationships by scoring one or more paths of direct links between two indirectly related nodes by analyzing weights associated with direct links that make up the path between the nodes using at least a damping factor; wherein one or more values resulting from the proximity analysis is stored in an index prior to searching by an end user and is used to determine an object's importance;

using the values of the proximity analysis to compare the object's importance to other objects in a pool of objects identified by using at least a word search for purposes of ranking search results; wherein said ranking of search results is also influenced by the number of times an object is visited;

displaying one or more determined hyperjump data *of objects in the pool.*

16. A method for navigating documents on the world wide web, comprising:

choosing a document;

identifying documents that have a direct relationship to the chosen document;

locating documents that have an indirect relationship to the chosen document identifying Universal Resource Locators for the documents, wherein the identified Universal Resource Locators either point to or point away from the chosen document;

analyzing Universal Resource Locators, including the identified Universal Resource Locators, wherein Universal Resource Locators which have an indirect relationship to the chosen document are located, wherein the step of analyzing further comprises [cluster] analyzing the Universal Resource Locators for indirect relationships *using cluster links*; and

displaying a located document.

21. A method of displaying information about a network that has hyperjump data, comprising:

choosing a node;

accessing the hyperjump data;

identifying hyperjump data from within the accessed hyperjump data that has a direct reference to the chosen node;

determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node using the identified hyperjump data, wherein the step of determining comprises [cluster analyzing the hyperjump data] *non-semantically generating a set of candidate cluster links for nodes indirectly related to the chosen node using the hyperjump data, assigning weights to the candidate cluster links and deriving actual cluster links from the set of candidate cluster links based on the assigned weights*; and

displaying one or more determined hyperjump data.

22. A method for displaying information about a network that has hyperjump data, comprising:

choosing a node;

accessing the hyperjump data;

identifying hyperjump data from within the accessed hyperjump data that has a direct reference to the chosen node;

determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node [using] *by proximity indexing* the identified hyperjump data; and

displaying one or more determined hyperjump data, wherein the nodes are nodes in the network that may be accessed, the [hypejump] *hyperjump* data includes hyperjump links between nodes in the network, and the step of displaying comprises:

generating a source map using one or more of the determined hyperjump data, wherein the source map represents hyperjump links that identify the chosen node as a destination of a link, and wherein the method further comprises activating a link represented on the source map, wherein a user may hyperjump to a node represented as a node of the link.

23. A method for displaying information about a network that has hyperjump data, comprising:

choosing a node;

accessing the hyperjump data;

identifying hyperjump data from within the accessed hyperjump data that has a direct reference to the chosen node;

determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node using the identified hyperjump data, wherein the step of determining comprises proximity analyzing the identified hyperjump data; and

displaying one or more determined hyperjump data,

wherein the chosen node is an object stored in a database that has direct relationships with other objects in said database and said direct relationships relate to hyperlink relationships on the world wide web; and wherein the step of proximity analyzing comprises:

analyzing indirect relationships by scoring one or more paths of direct links between two indirectly related nodes by analyzing weights associated with direct links that make up the path between the nodes.

24. The method of claim 23, wherein said proximity analysis uses a damping factor and comprises:

determining a quantity of hyperlinks within an object and scoring at least one indirect relationship comprising at least four link lengths from the chosen node.

25. The method of claim 24, wherein one or more values resulting from the proximity analysis is stored prior to searching by an end user and is used to determine an object's importance; the method further comprising:

using the proximity analysis to compare the object's importance to other objects in a pool of objects identified by using at least a word search for purposes of ranking search results.

26. The method of claim 23, wherein the step of displaying is influenced by a number of times a web object is visited.

27. A method for visually displaying data related to a web having identifiable web pages and Universal Resource Locators with pointers, comprising:

choosing an identifiable web page;

identifying Universal Resource Locators for the web pages, wherein the identified Universal Resource Locators either point to or point away from the chosen web page;

analyzing Universal Resource Locators, including the identified Universal Resource Locators, wherein Universal Resource Locators which have an indirect relationship to the chosen web page are located, wherein the step of analyzing further comprises cluster analyzing the Universal Resource Locators for indirect relationships; and

displaying identities of web pages, wherein the located Universal Resource Locators are used to identify web pages,

wherein cluster analyzing for indirect relationships comprises generating and deriving cluster links, wherein said generating cluster links comprises scoring a path of direct link weights between two nodes relating to the world wide web using at least a quantity of hyperlinks within an object, and wherein said identification of web pages using the located Universal Resource Locators comprises using the located Universal Resource Locators as part of an analysis to determine a rank used to locate or display objects responsive to a search.

28. A method for visually displaying data related to a web having identifiable web pages and Universal Resource Locators with pointers, comprising:

choosing an identifiable web page;

identifying Universal Resource Locators for the web pages, wherein the identified Universal Resource Locators either point to or point away from the chosen web page;

analyzing Universal Resource Locators, including the identified Universal Resource Locators, wherein Universal Resource Locators which have an indirect relationship to the chosen web page are located, wherein the step of analyzing further comprises cluster analyzing the Universal Resource Locators for indirect relationships; and

displaying identities of web pages, wherein the located Universal Resource Locators are used to identify web pages,

wherein the step of displaying is influenced by a number of times a web object is visited and wherein the cluster analysis uses a damping factor.

29. The method of claim 27, wherein one or more values resulting from the cluster analysis is stored prior to searching by an end user and is used to determine an object's importance; the method further comprising:

using the cluster analysis to compare the object's importance to other objects in a pool of objects identified by using at least a word search for ranking search results.

30. A method for navigating documents on the world wide web, comprising: chooosing a document;

identifying documents that have a direct relationship to the chosen document;

locating documents that have an indirect relationship to the chosen document identifying Universal Resource Locators for the documents, wherein the identified Universal Resource Locators either point to or point away from the chosen document;

analyzing Universal Resource Locators, including the identified Universal Resource Locators, wherein Universal Resource Locators which have an indirect relationship to the chosen document are located, wherein the step of analyzing further comprises cluster analyzing the Universal Resource Locators for indirect relationships; and

displaying a located document,

wherein the step of cluster analyzing comprises:

generating and deriving cluster links wherein said generating comprises scoring a path of direct link weights between two nodes by recursively analyzing the set of direct links and using a damping factor,

and wherein said direct relationships to a chosen document comprise a reference from the chosen document to at least one of the identified documents.

31. The method of claim 16, wherein the step of displaying is influenced by a number of times a web object is visited.

32. A method for displaying information about a network that has hyperjump data, comprising:

choosing a node;

accessing the hyperjump data;

identifying hyperjump data from within the accessed hyperjump data that has a direct reference to the chosen node;

determining hyperjump data from within the accessed hyperjump data that has an indirect reference to the chosen node using the identified hyperjump data, wherein the step of determining comprises cluster analyzing the hyperjump data; and

displaying one or more determined hyperjump data,

wherein the step of cluster analyzing comprises:

generating deriving cluster links, wherein said generating comprises scoring a path of direct link weights between two indirectly related nodes by recursively analyzing the set of direct links, using a damping factor, and using a quantity of direct references within an object to another object; and

wherein said direct reference to a chosen document comprises a reference from the chosen document to another object.

* * * * *

ADDENDUM 3

Final Written Decision – 35 U.S.C. § 318(a) and 37 C.F.R. § 42.73,
Facebook, Inc. et al. v. Software Rights Archive, LLC, Case IPR2013-00478
(P.T.A.B. February 2, 2015) (Paper 58)

1 – 42

Trials@uspto.gov
352.272.7822

Paper 58
Entered: February 2, 2015

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

FACEBOOK, INC., LINKEDIN CORP., and TWITTER, INC.,
Petitioner,

v.

SOFTWARE RIGHTS ARCHIVE, LLC,
Patent Owner.

Case IPR2013-00478
Patent 5,544,352

Before SALLY C. MEDLEY, CHRISTOPHER L. CRUMBLEY, and BARBARA A. PARVIS, *Administrative Patent Judges.*

CRUMBLEY, *Administrative Patent Judge.*

FINAL WRITTEN DECISION
35 U.S.C. § 318(a) and 37 C.F.R. § 42.73

I. BACKGROUND

A. Introduction

On July 30, 2013, Facebook, Inc., LinkedIn Corp., and Twitter, Inc. (collectively "Petitioner") filed a Petition requesting an *inter partes* review of claims 26, 28–30, 32, 34, and 39 of U.S. Patent No. 5,544,352 (Ex. 1001, "the '352 patent"). Paper 1 ("Pet."). On February 3, 2014, we instituted

trial on all challenged claims, on certain of the grounds of unpatentability alleged in the Petition. Paper 17 ("Decision to Institute" or "Inst. Dec.").

After institution of trial, Software Rights Archive, LLC ("Patent Owner"), filed a Patent Owner Response ("PO Resp."). Paper 34. Petitioner also filed a Reply. Paper 43 ("Reply").

A consolidated oral hearing for IPR2013-00478, IPR2013-00479, IPR2013-00480, and IPR2013-00481, each involving the same Petitioner and the same Patent Owner, was held on October 30, 2014. The transcript of the consolidated hearing has been entered into the record. Paper 57, "Tr."

We have jurisdiction under 35 U.S.C. § 6(c). This Final Written Decision is issued pursuant to 35 U.S.C. § 318(a) and 37 C.F.R. § 42.73.

Petitioner has shown by a preponderance of the evidence that claims 26, 28–30, 32, 34, and 39 of the '352 patent are unpatentable.

B. Related Proceedings

Petitioner and Patent Owner both indicate that the '352 patent is involved in the following co-pending district court proceedings: *Software Rights Archive, LLC v. Facebook, Inc.,* Case No. 12-cv-3970; *Software Rights Archive, LLC v. LinkedIn Corp.,* Case No. 12-cv-3971; and *Software Rights Archive, LLC v. Twitter, Inc.,* Case No. 12-cv-3972, each pending in the United States District Court for the Northern District of California. Pet. 1; Paper 8, Patent Owner's Mandatory Notice, 2.

In addition, we instituted trial on Petitioner's petitions on related patents including: (1) IPR2013-00479 and IPR2013-00480, *inter partes* reviews of U.S. Patent No. 5,832,494 (the "'494 patent"); and (2) IPR2013-00481, an *inter partes* review of U.S. Patent No. 6,233,571 (the "'571 patent"). The '352 patent issued from the parent of the application that

issued as the ’494 patent. The ’571 patent issued from an application that was a divisional of the application that issued as the ’494 patent. The ’352 patent was the subject of Reexamination No. 90/011,010.

C. The ’352 patent

The ’352 patent relates to computerized research on databases. Ex. 1001, 1:7–11. The ’352 patent discloses that it improves search methods by indexing data using proximity indexing techniques. *Id*. at 3:42–55. According to the ’352 patent, proximity indexing techniques generate a quick-reference of the relations, patterns, and similarity found among the data in the database. *Id*. at 3:53–55.

Figure 2 of the ’352 patent illustrates the high-level processing of software for computerized searching (*Id.* at 8:7–8) and is reproduced below:

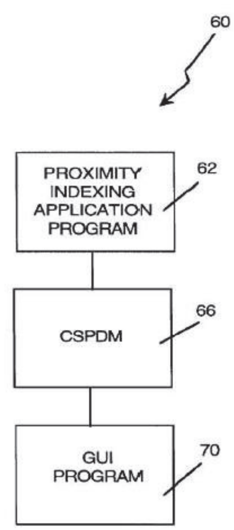


Fig. 2

Figure 2 depicts software system 60 comprising Proximity Indexing Application Program 62, Computer Search Program for Data Represented by Matrices (“CSPDM”) 66, and Graphical User Interface (“GUI”) program 70.

Id. at 10:53–60.

Processing of software system 60 begins with Proximity Indexing Application Program 62 indexing a database. *Id.* at 11:4–5. Then, CSPDM 66 searches the indexed database and retrieves requested objects. *Id.* at 11:6–10. CSPDM 66 relays the retrieved objects to GUI program 70 to display on a display. *Id*. at 11:10–13.

Software system 60 runs on a computer system comprising, for example, a processor of a personal computer. *Id.* at 9:39–44. The system comprises a display, which displays information to the user. *Id.* at 10:4–7. Exemplary displays include computer monitors, televisions, LCDs, or LEDs. *Id*.

The processor is connected to a database to be searched. *Id*. at 9:46–47. The database contains cases—also called full textual objects—that contain citations to other objects within the database. *Id*. at 12:1–10. Each full textual object is assigned a number corresponding to its chronological order in the database. *Id*.

The '352 patent discloses that any two textual objects in the database may be related through a number of "patterns." *Id*. at 12:31–32. For example, object A may cite B, or the two objects may cite the same object C. *Id*. at 12:46–61. The Proximity Indexing Application (discussed above) applies algorithms to these relationships to create a matrix of pattern vectors that represent the relationships between the various objects in the database. *Id*. at 12:62–13:3, 14:18–20. The CSPDM is used to search the indexed database. *Id*. at 14:20–21.

D. Illustrative Claim

Of the challenged claims, only claim 26 is independent, whereas claims 28–30, 32, 34, and 39 depend, directly or indirectly, from claim 26. Claim 26 is illustrative of the claimed subject matter and is reproduced below:

> 26. A non-semantical method for numerically representing objects in a computer database and for computerized searching of the numerically represented objects in the database, wherein direct and indirect relationships exist between objects in the database, comprising:
>
> marking objects in the database so that each marked object may be individually identified by a computerized search;
>
> creating a first numerical representation for each identified object in the database based upon the object's direct relationship with other objects in the database;
>
> storing the first numerical representations for use in computerized searching;
>
> analyzing the first numerical representations for indirect relationships existing between or among objects in the database;
>
> generating a second numerical representation of each object based on the analysis of the first numerical representation;
>
> storing the second numerical representation for use in computerized searching; and
>
> searching the objects in the database using a computer and the stored second numerical representations, wherein the search identifies one or more of the objects in the database.

Ex. 1001, 35:28–54.

E. The Prior Art References Upon Which Trial Was Instituted

Yahiko Kambayashi et al., *Dynamic Clustering Procedures for Bibliographic Data*, Kyoto Univ., Dep't of Inf. Sci., 90–99 (1981) ("Kambayashi") (Ex. 1004).

Colin F.H. Tapper, *Citation Patterns in Legal Information Retrieval*, 3 DATENVERARBEITUNG IM RECHT 249–75 (1976) ("Tapper 1976") (Ex. 1005).

Colin Tapper, *The Use of Citation Vectors for Legal Information Retrieval*, 1 J. OF LAW AND INFO. SCI. 131–61 (1982) ("Tapper 1982") (Ex. 1006).

Edward A. Fox, *Characterization of Two New Experimental Collections in Computer and Information Science Containing Textual and Bibliographic Concepts* (Sept. 1983) (Ph.D. dissertation, Cornell Univ. Dep't of Comp. Sci.) ("Fox Collection") (Ex. 1007).

Edward A. Fox, *Some Considerations for Implementing the SMART Information Retrieval System under UNIX* (Sept. 1983) (Ph.D. dissertation, Cornell Univ. Dep't of Comp. Sci.) ("Fox SMART") (Ex. 1008).

Edward A. Fox, *Extending the Boolean and Vector Space Models of Information Retrieval with P-Norm Queries and Multiple Concept Types* (Aug. 1983) (Ph.D. dissertation, Cornell Univ. Dept. of Comp. Sci.) ("Fox Thesis") (Ex. 1009).

The parties do not dispute the prior art status of the references.

F. The Pending Grounds of Unpatentability

Reference(s)	Basis	Claims instituted
Kambayashi	§ 102	26, 28–30, 32, 39
Fox Thesis, Fox SMART, and Fox Collection	§ 103	26, 28–30, 32, 34, 39
Tapper 1976 and Tapper 1982	§ 103	26, 28–30, 32, 34, 39

II. ANALYSIS

A. Claim Construction

1. Principles of Law

Petitioner asserts, and Patent Owner does not dispute, that the ’352 patent expired on August 6, 2013. Pet. 6. The Board’s interpretation of the claims of an expired patent is similar to that of a district court’s review. *See In re Rambus, Inc.*, 694 F.3d 42, 46 (Fed. Cir. 2012). We, therefore, are guided by the principle that the words of a claim “are generally given their ordinary and customary meaning,” as understood by a person of ordinary skill in the art in question at the time of the invention. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–13 (Fed. Cir. 2005) (en banc) (citation omitted). “In determining the meaning of the disputed claim limitation, we look principally to the intrinsic evidence of record, examining the claim language itself, the written description, and the prosecution history, if in evidence.” *DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.,* 469 F.3d 1005, 1014 (Fed. Cir. 2006) (citing *Phillips*, 415 F.3d at 1312–17). There is a “heavy presumption,” however, that a claim term carries its ordinary and customary meaning. *CCS Fitness, Inc. v. Brunswick Corp.*, 288 F.3d 1359, 1366 (Fed. Cir. 2002) (citation omitted).

2. Overview of the Parties' Positions

In the Decision to Institute, we found it instructive to construe the claim terms *direct relationships*, *indirect relationships*, *pool-similarity searching*, and *pool-importance searching*. Inst. Dec. 10–14. Our constructions are set forth in the table below.

Claim Term or Phrase	Construction
direct relationships	"relationships where one object cites to another object" Inst. Dec. 13.
indirect relationships	"relationships where at least one intermediate object exists between two objects and where the intermediate object(s) connect the two objects through a chain of citations" Inst. Dec. 13.
pool-similarity searching	"identifying at least one object based on degree of similarity to a selected pool of objects" Inst. Dec. 14.
pool-importance searching	"identifying at least one object based on the importance of the object to a selected pool of objects" Inst. Dec. 14.

Petitioner does not challenge any of our constructions. Reply 1–2. Patent Owner appears to agree with many of our constructions, and states that it uses our constructions for the purpose of evaluating patentability of the challenged claims of the '352 patent. PO Resp. 12–14. Based on the complete record now before us, we discern no reason to change our prior constructions.

Additionally, Patent Owner addresses the following phrases or terms: 1) *objects in a computer database*; 2) *computerized searching*; 3) *non-semantical method*; 4) *some indirect relationships are weighed more heavily than other indirect relationships*; and 5) *relationships exist between or among subsets of objects*, which are discussed below. *Id*. at 9–14.

Petitioner's Reply further addresses *database* and *numerical representation*, but otherwise does not contest Patent Owner's proposed constructions of these terms. Reply 2.

3. numerical representation

Patent Owner's Response does not proffer an explicit construction of *numerical representation*, but appears to interpret the term to exclude strings that may include letters. PO Resp. 21 (distinguishing prior art as having "non-numerical character strings). At oral argument, Patent Owner confirmed that its construction of *numerical representation* is something "represented only by digits," or in other words "expressed by numbers, not by letters." Tr. 85.

Petitioner responds that *numerical* includes "any representation of binary or digital data that can be processed and analyzed by a computer," and means simply "of or relating to numbers." Reply 1; Tr. 13. Petitioner's construction is, therefore, not limited to representations consisting only of numbers. At oral argument, Petitioner argued that the inclusion of a single number into a string is sufficient to make that string a *numerical representation*. Tr. 25.

Petitioner's proffered construction is overly broad and unsupported by the specification. While one dictionary definition of *numerical* is "of or relating to a number or series of numbers," it may also refer to "expressed in or counted by numbers." THE AMERICAN HERITAGE DICTIONARY OF THE ENGLISH LANGUAGE (2000) (Ex. 3001); *see also* COLLINS ENGLISH DICTIONARY (2000) (Ex. 3002) ("measured or expressed in numbers").

The specification of the '352 patent uses *numerical* consistent with this latter interpretation. In the Initial Extractor Subroutine, the "full textual

objects" of the database are numbered "with Arabic numbers from 1 through n." Ex. 1001, 14:49–50. These numbers are used to create vectors and matrices, which are then run through various algorithms such as the Opinion Patterner Subroutine. *Id*. at 14:55–15:22. "Numerical factors" are then "calculated" to determine "values." *Id*. at 15:19–22; 18:63–67. This emphasis on calculation, values, and on processing by computer algorithms, leads us to conclude that *numerical representation*, as used in the '352 specification, must refer to solely numbers, so that a computer can process the representations using mathematical algorithms.

Petitioner's attempt to link the *numerical representation* of the specification to the West "key number" system is unpersuasive. Reply 2. While the specification of the '352 patent does discuss the key number system, and such "key numbers" include letters, there is no indication that the patentee intended to link the *numerical representation* of the claims to the West key number system discussed—and distinguished—in the background portion of the specification. Ex. 1001, 2:38–43 ("such a numbering process is subjective and is prone to error").

Nor do we find persuasive Petitioner's argument that *numerical* is somehow distinct from "numeric," in that the latter term means only numbers but the former may encompass letters. Tr. 13. Not only was this argument advanced for the first time at oral hearing,[1] but it is unsupported by any evidence of record. Indeed, the two terms are used interchangeably in dictionary definitions. *See* Ex. 3001 (entry for "numerical also numeric"); Ex. 3002 (entry for "numerical or numeric").

[1] Our Rules do not permit arguments to be raised for the first time at oral hearing. 37 C.F.R. § 42.70(a) (permitting oral argument only on "an issue raised in a paper.").

For these reasons, we construe *numerical representation* as "representation consisting exclusively of numbers or a set of numbers."

4. *objects in a computer database and computerized searching*

Patent Owner addresses the terms *objects in a computer database* and *computerized searching*, both of which appear in claim 26. PO Resp. 9–11. Rather than proffer a construction for either term, however, Patent Owner discusses the general concept of computerized searching in the '352 patent. *Id*. It is not clear what construction Patent Owner wishes us to adopt, and we are not persuaded that either term requires an explicit construction.

5. *non-semantical method*

Patent Owner asks that we interpret *non-semantical method* to mean "a method that uses the direct relationships between one database object and another and does not otherwise account for words and phrases in a textual object." PO Resp. 11. We note that Petitioner raised a similar construction in the Petition, but the Board declined to construe the term expressly in the Decision to Institute. Pet. 6–7; Inst. Dec. 11. We also note that we adopted a similar construction for "non-semantically" in the related case IPR2013-00481. We consider the proffered construction to be reasonable and consistent with the specification of the '352 patent, and adopt it herein.

6. *some indirect relationships are weighed more heavily than other indirect relationships*

Patent Owner asks that we construe this phrase as "some *types* of indirect relationships are weighed more heavily than others." PO Resp. 13 (emphasis in original). To support this interpretation, Patent Owner cites the specification of the '352 patent, which discloses that the different "patterns," which include direct and indirect relationships, are assigned various weights.

Ex. 1001, 13:34–38. Petitioner does not dispute this construction. We consider the proffered construction to be reasonable and consistent with the specification of the ’352 patent, and adopt it herein.

7. relationships exist between or among subsets

Patent Owner does not set forth an express construction for this phrase which appears in claim 34, but instead states that the “relationships” are the direct and indirect relationships of claim 26, and that subsets are a portion of a textual object. PO Resp. 13–14. Patent Owner has not persuaded us that an express construction of this phrase is necessary.

B. Anticipation of Claims 26, 28–30, 32, and 39 by Kambayashi

We instituted trial to determine whether claims 26, 28–30, 32, and 39 are unpatentable under 35 U.S.C. § 102 as anticipated by Kambayashi. Dec. Inst. 19–20. To establish anticipation, Petitioner must prove that each and every element in a claim, arranged as is recited in the claim, may be found in a single prior art reference. *Net MoneyIN, Inc. v. VeriSign, Inc.*, 545 F.3d 1359, 1369 (Fed. Cir. 2008); *Karsten Mfg. Corp. v. Cleveland Golf Co.*, 242 F.3d 1376, 1383 (Fed. Cir. 2001). We determine that Petitioner has not shown by a preponderance of the evidence that claims 26, 28–30, 32, and 39 are unpatentable as anticipated by Kambayashi.

1. Kambayashi

Kambayashi describes a method for clustering, which is said to be “an important tool for efficient retrieval of documents in bibliographic database systems.” Ex. 1004, Abstract. The reference discloses the creation of “Direct Reference Matrix R,” defining direct reference as “when a paper A refers to a paper B.” *Id*. at 91–92. Kambayashi also discloses a set of pairs

(ID, IDF) where ID and IDF are the identification codes of the papers and (ID, IDF) means that paper ID cites paper IDF. *Id*. at 93.

Kambayashi also discloses the creation of two secondary matrices, "Bibliographic Coupling Matrix B" (papers with one or more citation in common) and "Co-citation Matrix C" (citations frequently cited together). *Id*. at 92. These secondary matrices consist of vectors derived from the (ID, IDF) pairs noted above. *Id*. at 93–34. A "Similarity Matrix S" may then be created via weighted summation of matrices R, B, and C. *Id*. at 92.

2. Claim 26

We focus our analysis herein on two steps required by the method of claim 26: "creating a first numerical representation for each identified object in the database based upon the object's direct relationship with other objects in the database," and "analyzing the first numerical representations for indirect relationships existing between or among objects in the database." Petitioner contends that Kambayashi discloses both of these steps in two alternative embodiments.

First, Petitioner directs us to Kambayashi's disclosure of (ID, IDF) pairs, and their use in creating the B and C matrices. Petitioner asserts that deriving the (ID, IDF) pairs may be considered to be creating a first numerical representation, and that they represent direct relationships between documents ID and IDF. Pet. 24–25; Tr. 22. The (ID, IDF) pairs are then analyzed for indirect relationships, leading to B and C matrices which Petitioner contends are second numerical representations. Pet. 25–26.

Second, Petitioner identifies Kambayashi's Direct Reference Matrix R as a first numerical representation that represents direct relationships. Pet. 24–25; Tr. 22. As noted above, Matrix R is used—along with matrices B

and C—to generate a Similarity Matrix S. Petitioner contends that if Matrix R is considered to be the first numerical representation, then Matrix S would be a second numerical representation within the scope of claim 26. Tr. 22.

Upon review of the disclosure of Kambayashi, we find neither of these arguments persuasive. First, in the case of the (ID, IDF) pairs that are used to derive the B and C matrices, Patent Owner argues that ID and IDF are *strings* that contain letters. PO Resp. 21. This is supported by the disclosure of Kambayashi, which discloses identification codes such as EVER7404 and GARDL7710. Ex. 1004, 96. The testimony of Dr. Jacobs, Patent Owner's declarant, explains how Kambayashi's source database shows that identification codes begin with the first four letters of the first author's name. Ex. 2113 ¶¶ 100–103 (citing Ex. 2023). Because, as discussed above, the proper construction of *numerical representation* is a representation that contains only numbers, Kambayashi's (ID, IDF) pairs cannot be the first numerical representation of claim 26.

Nor do we find that the Direct Reference Matrix R / Similarity Matrix S system of Kambayashi meets the limitations of claim 26. Petitioners assert, and we agree, that Matrix R is an array of numbers that represents direct relationships between the objects in the Kambayashi database. Pet. 24–25; Ex. 1004, 92 (values r_{ij} of matrix R are either 0 or 1). This conclusion is supported by the fact that, in order to derive Matrix S, Matrix R is multiplied by a constant w_R. Ex. 1004, 92. It would not make sense to multiply a matrix containing strings of letters by a constant.

Matrix S, however, cannot be the second numerical representation of claim 26. The claim requires that the representation be created by "analyzing the first numerical representations for indirect relationships

existing between or among objects in the database." Matrix S, however, is generated according to the following formula:

$S = w_R*R+w_B*B'+w_C*C'$ (*Id.*)

Patent Owner argues that multiplying matrix R by a constant is not *analyzing*, as that term is used in claim 26. PO Resp. 21. We agree. While Matrix S does take into account indirect relationships between objects, those relationships are not derived from an analysis of Matrix R (the first numerical relationship). Rather, the indirect relationships are accounted for in Matrix S by the inclusion of Matrices B (which tracks bibliographic coupling) and C (which tracks co-citation). *Id*. The indirect relationships reflected in the B and C matrices, in turn, are not derived from Matrix R, but rather from the (ID, IDF) pairs, which we have determined above cannot be the first numerical relationship. For these reasons, Matrix S of Kambayashi does not meet the second numerical representation limitation of claim 26, because it is not generated by analyzing the first numerical representation.

We, therefore, conclude that Kambayashi fails to disclose an embodiment having all elements of claim 26, as arranged in the claim. Kambayashi does not anticipate claim 26.

3. Dependent Claims

The remaining instituted claims all depend, directly or indirectly, from claim 26, and incorporate claim 26's requirements of a first numerical representation, and second numerical representation. We, therefore, find that Kambayashi does not anticipate the dependent claims, for the same reasons discussed above with respect to claim 26.

C. Obviousness of Claims 26, 28–30, 32, 34, and 39 Over Fox Papers

We instituted trial to determine whether claims 26, 28–30, 32, 34, and 39 are unpatentable under 35 U.S.C. § 103 as having been obvious over the combined disclosures of Fox Thesis, Fox SMART, and Fox Collection (collectively, "the Fox Papers"). Inst. Dec. 14–19. In support of the asserted ground of unpatentability, Petitioner sets forth the teachings of the cited prior art, provides detailed claim charts, and cites to the declaration of Dr. Fox (Ex. 1016 ¶¶ 68–145), explaining how each limitation is taught in the cited prior art combination. Pet. 9–23.

The claim chart persuasively reads all elements of each of claims 26, 28–30, 32, 34, and 39 onto the teachings of the Fox Papers, taken together. Despite the counter-arguments in Patent Owner's Response, and the evidence cited therein, which we have also considered, Petitioner has shown by a preponderance of the evidence that each of claims 26, 28–30, 32, 34, and 39 of the '352 patent are unpatentable, under 35 U.S.C. § 103, as they would have been obvious over the combination of Fox Thesis, Fox Collection, and Fox SMART.

1. Fox Thesis

Fox Thesis describes improving query and document representation schemes for information retrieval. Ex. 1009, 261. In particular, useful types of bibliographic data are incorporated into a model to test clustering and retrieval functions. *Id.* at 164. Bibliographic connections between articles are illustrated for an exemplary set "O" of documents, which are represented by letters A through G. *Id.* at 165–66, Fig. 6.2. This exemplary set "O" includes direct and indirect citation references. *Id.* at 166–67, Table 6.2.

Based on the reference pattern for a set of documents, Fox Thesis describes deriving various measures of the interconnection between the documents. *Id.* at 166. For example, weights are assigned "based upon integer counts" for bibliographically coupled documents. *Id.* at 167. Citation submatrices represent reference or citation information. *Id.* at 169. For example, submatrix *bc* represents bibliographically coupled reference information and submatrix *cc* represents co-citation reference information. *Id.* at 169–72, Figs. 6.3–6.5.

2. Fox SMART

Fox SMART describes the System for Mechanical Analysis and Retrieval of Text ("SMART") as a project for designing a fully automatic document retrieval system and for testing new ideas in information science. Ex. 1008, 3. Fox SMART describes the computer system used to implement the experiments described in the Fox Thesis. Ex. 1016, ¶ 27. The software components of SMART are implemented in the C Programming Language and run under the UNIX™ operating system on a VAX™ 11/780 computer. Ex. 1008, 1, 4.

In SMART, an automatic indexing component constructs stored representations of documents. *Id.* at 3. Bibliographic information is used to enhance document representations. *Id.* at 29. The SMART system may process basic raw data, such as an exemplary N collection of articles and citation data describing which articles are cited by others. *Id.* at 29–30. Data is entered into the SMART system as a set of tuples $\{(d_i, d_j) | d_i \rightarrow d_j\}$ which describe the cited and citing documents, as well as the direction of citation. *Id.* at 29. The exemplary input data also includes indirect citation relationships, such as bibliographic coupled and co-citation relationships.

Id. at 30–32. These relationships are used to create extended vectors which can then be clustered and searched to aid document retrieval. *Id*. at 29.

3. Fox Collection

Fox Collection describes collections of data which are said to be useful for investigating the interaction of textual and bibliographic data in retrieval of documents. Ex. 1007, 1. According to the testimony of Dr. Edward Fox, Fox Collection was originally part of the same work as Fox Thesis and Fox SMART, and describes the manner in which the data sets were obtained and processed prior to their use in the Fox SMART experiments. Ex. 1016 ¶ 27.

According to Fox Collection, the experiments were performed on a collection of bibliographic records (title, abstract, author, keywords, etc.) from the *Communications of the ACM*, termed the "CACM collection." Ex. 1007, 14.[2] Two individuals then examined printed copies of the articles referenced by the CACM bibliographic records, and citation data was obtained from the articles and entered into a set Raw_data. *Id*. The citation data contained pairs of identifiers (citing, cited) which were the document id numbers ("dids") of the citing record and record it cites. *Id*. From this Raw_data matrix, secondary matrices such as bc (bibliographic coupling) and cc (co-citation) were derived computationally. *Id*. at 14–16.

---

[2] Fox Collection also discusses an ISI Collection, but in his Reply Declaration Dr. Fox explains that he cites the ISI collection to "emphasize findings in the prior art about the value of using co-citation data (a non-semantic indirect relationship) in information retrieval, not to fully address all the elements of claims. . . . For the sake of simplicity, the Board should focus on the methodology given in Fox Papers, and the examples of their use with the CACM Collection." Ex. 1030, ¶ 6.

4. Claim 26

Petitioner's claim chart persuasively reads all elements of claim 26 onto the combined teachings of Fox Thesis, Fox SMART, and Fox Collection. Pet. 9–23 (citing Ex. 1007, 14–15, 43, 48; Ex. 1008, 3, 12–13, 16, 18, 25–27, 29–33, 36, 38–39, 41–43, 53; Ex. 1009, 17, 19, 179, 181–82, 195, 199, 203, 211; 1016 ¶¶ 71–108, 122–131). For instance, the combination of Fox Thesis, Fox SMART, and Fox Collection teaches "marking objects in the database" and "creating a first numerical representation for each identified object in the database based upon the object's direct relationship with other objects in the database," as recited in claim 26. In particular, Fox Collection teaches assigning document identification numbers ("dids") to the articles in the CACM collection, which is "marking objects in the database." Ex. 1007, 14. Printed copies of each article with a bibliographic entry in the CACM collection then are reviewed manually, to obtain bibliographic subvectors in the form "Raw_data (cited, citing)." *Id*. This is a first numerical representation created based on the direct relationship between the "cited, citing" pair of bibliographic records.

The combination of Fox Thesis, Fox SMART, and Fox Collection also teaches "analyzing the first numerical representations for indirect relationships existing between or among objects in the database" and "generating a second numerical representation of each object based on the analysis of the first numerical representation," as recited in claim 26. Fox SMART teaches that direct relationships may be represented by tuples called "CITED," which contain a citing document, a cited document, and the direction of the citation. Ex. 1008, 29. These tuples are then processed to

construct submatrices such as *bc* and *cc*, which contain numbers representing indirect relationships. *Id*. at 30–32 ("construct BC by counting the number of identical tuples of C"). Dr. Fox testifies that the CITED tuples of Fox SMART refer to the Raw_data derived from the CACM collection. Ex. 1016 ¶ 124. Because these *bc* and *cc* submatrices are numerical representations, and are generated from the first numerical representations CITED which are based on direct relationships, we find that the Fox Papers together teach "generating a second numerical representation of each object based on the analysis of the first numerical representation."

a. Combination of References

As to whether Petitioner has satisfied the requirements for combining the teachings of Fox Thesis, Fox SMART, and Fox Collection, we determine that Petitioner has articulated sufficient reasoning with a rational underpinning as to why one of ordinary skill in the art would have combined the retrieval systems taught in Fox Thesis, Fox SMART, and Fox Collection. *See KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007) (citing *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)). Dr. Fox states that the three publications arose from the same thesis project, and were originally one document. Ex. 1016 ¶ 70. Furthermore, Dr. Fox notes that Fox Thesis "explain[s] the method and experimental results of [his] information retrieval work," Fox SMART "detail[s] the updated SMART computer system used to execute the experiments," and Fox Collection "describes how the data sets were obtained and processed prior to being used in the experiments." *Id*. We give Dr. Fox's statement that one of skill in the art would have been motivated to combine the references because they "describe a complete project with its underlying system and data" (*id*.)

substantial weight, because it is consistent with the considerable overlap in the disclosures of the Fox Papers and their internal references to one another. *See, e.g.,* Ex. 1009, 343 (Fox Thesis cites to Fox SMART); Ex. 1008, 84 (Fox SMART cites to Fox Thesis).

b. Patent Owner's Counterarguments

We have considered Patent Owner's counterarguments but do not find them persuasive. Patent Owner contends that various elements of claim 26 are not taught or suggested by the Fox Papers in combination. First, Patent Owner argues that the Fox Papers do not teach a computer database in which direct and indirect relationships exist between objects in the database. PO Resp. 17. Because the databases of the Fox Papers are bibliographic databases, they contain certain information about documents such as title, abstract, author, and publication date. *See, e.g.,* Ex. 1007, 14 (describing CACM database). The Fox databases do not contain the full documents, meaning that the databases do not contain the portions of the documents that cite to other documents. As such, Patent Owner argues, the databases cannot have objects having direct and indirect relationships, as required by claim 26. PO Resp. 17–18.

We have construed *direct relationships* to mean "relationships where one object cites to another object." Based on this construction, the bibliographic records of Fox Collection's CACM database do not have direct relationships, because they do not contain cites to one another. It is only after the full documents—which are not in the database—are manually reviewed, and the first numerical representation (Raw_data) is entered, that the database contains objects that have direct relationships. Claim 26, however, requires that direct and indirect relationships exist between objects

in the database first, prior to creating a first numerical representation. In other words, the Raw_data of the Fox Collection CACM database cannot be both the objects that have relationships, as well as the first numerical representation of those relationships.

Petitioner contends, however, that it would have been "trivial and obvious" to modify the databases of the Fox Papers to contain full text documents. Reply 10. Dr. Fox's testimony supports this argument, noting that if storage resources allowed storage of the full text of documents, this would have been understood as preferable. Ex. 1016 ¶¶ 76, 89. We credit Dr. Fox's testimony on this point, as it is consistent with the disclosure of Fox Thesis that "some [information retrieval] systems store the full text of the various documents." Ex. 1009, 6. Fox Thesis adds that full text permits users to "locate documents of interest," as well as "retrieve and/or examine paragraphs, passages, sentences, or single word occurrences (in context)." *Id*. These extra capabilities are described as "straightforward generalizations of document retrieval methods." *Id*.

We, therefore, conclude that the Fox Papers suggested to one of ordinary skill in the art at the time of the invention the modification of the Fox databases to include full text documents. With such a modification, the databases would contain, as objects, the full text documents. Therefore, even prior to generation of the Raw_data, the database would contain objects that have direct and indirect relationships due to their citation of one another. Patent Owner's argument to the contrary is unpersuasive.

In the same vein, Patent Owner argues that the Fox Papers do not teach "creating a first numerical representation for each identified object in the database based upon the object's direct relationship with other objects in

the database." PO Resp. 31. Because, for example, the Raw_data disclosed in the Fox Collection is derived from documents that are not in the CACM database, but rather compiled from full text printed versions of the documents, Patent Owner argues that Raw_data is not *based on* the object's direct relationship with other objects. *Id*. at 32.

We find this argument unpersuasive for the same reasons outlined above for the objects limitation. The Fox Papers suggest inclusion of full text documents in the databases. With such a modification, the databases would have—even prior to creation of Raw_data—objects with direct relationships. The subsequently-created Raw_data relation would be based on those objects, thus satisfying the first numerical representation element of claim 26.

Patent Owner's remaining contentions relate to whether the Petitioner has satisfied the requirements for combining the teachings of Fox Thesis, Fox SMART, and Fox Collection. For example, Patent Owner contends that the systems disclosed in the individual Fox Papers are "narrowly tailored" and would not have been combined merely because of their common authorship. PO Resp. 26.

As indicated above, we determine that Petitioner has articulated sufficient reasoning with a rational underpinning as to why one of ordinary skill in the art would have combined the retrieval systems taught in Fox Thesis, Fox SMART, and Fox Collection. *See KSR*, 550 U.S. at 398. For instance, Dr. Fox wrote each of Fox Thesis, Fox SMART, and Fox Collection. See Ex. 1009, i; Ex. 1008, 1; Ex. 1007, 1.

Patent Owner also contends that the Raw_data relation of Fox Collection could not be combined with the CITED tuples of Fox SMART,

because they are "fundamentally incompatible." PO Resp. 27. In support of this argument, Dr. Jacobs testifies, for example, that CITED does not describe using document ids ("dids") while Raw_data does. Ex. 2113 ¶¶ 170–171. Dr. Fox testifies to the contrary, stating that the CITED tuples of Fox SMART specifically refer to the Raw_data derived from the CACM collection. Ex. 1016 ¶ 124. We give Dr. Fox's testimony on this point substantial weight. Our determination is not only due to Dr. Fox's personal knowledge of the Fox Papers, but also supported by the descriptions of Raw_data and CITED in the references. The references indicate that both Raw_data and CITED contain pairs of document identifiers, with the sole difference being that CITED also contains a third data element that signifies the direction of the citation. Furthermore, while the description of CITED in Fox SMART is silent as to document ids, other portions of the document discuss dids which are an "index in range 1 . . . N." Ex. 1008, 36. We do not consider the combination of Raw_data with CITED, or the combination of the systems of Fox Collection, Fox SMART, and Fox Thesis, to be beyond the level of ordinary skill in the art.

Patent Owner further contends that using indirect relationships in a computerized search system would not have been predictable at the time of the invention of the '352 patent. PO Resp. 50. Patent Owner's contention is based on its view that the combined teachings of Fox Thesis, Fox SMART, and Fox Collection are not sufficient because they do not teach computerized searching of an electronic database. PO Resp. 54; *see also* Tr. 49 ("[T]he Fox papers by themselves don't get you there . . . every one . . . is directed to printed articles, not an electronic database."). According to Patent Owner, the prior art cited by Petitioner teaches experiments that are

not directed to a computer database, "but rather are directed toward limited experimentation with bibliographic relationships existing among paper documents." PO Resp. 1.

We disagree with Patent Owner. For example, Fox SMART teaches an implementation in which software components of SMART are implemented in the C Programming Language and run under the UNIX™ operating system on a VAX™ 11/780 computer. Ex. 1008, 1, 4. In SMART, an automatic indexing component constructs stored representations of documents. *Id*. at 3. In light of the various teachings of Fox Thesis, Fox SMART, and Fox Collection discussed herein, we determine that Fox Thesis, Fox SMART, and Fox Collection, taken together, teach or suggest computerized searching of an electronic database.

Patent Owner also contends that the inclusion of indirect relationships into search "degrades results," and therefore provides a teaching away from the invention. PO Resp. 50. As Patent Owner acknowledges, its evidence of degraded results does not teach away from the *combination* of the Fox Papers, but rather from the *modification* of the teachings of the Fox Papers to incorporate "an electronic database that has references to the objects in the database." Tr. 49–50. We found above, however, that the Fox Papers teach this feature. In addition, to the extent modification of the Fox Papers is necessary to meet claim 26, we have found that modification is expressly suggested by the Fox Papers themselves. The record is insufficient to establish a teaching away.

Patent Owner also asserts objective indicia of non-obviousness, focusing on Google's search engine using its PageRank algorithm. PO Resp. 56–60. As an initial matter, Patent Owner's contentions again appear

to be based on its view that the combined teachings of Fox Thesis, Fox SMART, and Fox Collection are not sufficient because they do not teach computerized searching of an electronic database. *Id*. at 58 ("Link analysis technology applied to the Web, as claimed in the '352 patent and embodied in PageRank, satisfied a long felt need for improved computerized search." (citation omitted)); Tr. 60–61 ("[I]t certainly wouldn't have been obvious to one of ordinary skill based on Fox's work to extend these ideas from this paper collection to electronic databases."). For the reasons discussed above, we disagree with Patent Owner's view and determine that Fox Thesis, Fox SMART, and Fox Collection, taken together, teach or suggest computerized searching of an electronic database.

Furthermore, we note that Patent Owner has not shown that the asserted success of a commercial embodiment of the '352 patent actually resulted from features recited in the claims of the '352 patent. Patent Owner has not provided sufficient evidence to support a nexus between claim 26 and the Google PageRank algorithm. Because Patent Owner has failed to provide the source code of PageRank, or any other detailed information beyond publicly-available, generalized hearsay statements about Google's search (Ex. 2050), the record is insufficient to prove that PageRank uses the method of claim 26.

Even if PageRank's algorithm incorporates the method of claim 26, we cannot determine that Google's success is due to the method of claim 26, as opposed to other elements of the algorithm. Patent Owner's declarant Dr. Amy N. Langville conceded that the Google search technology involves a combination of link analysis (non-semantic) and semantic searching, whereas claim 26 recites a non-semantical method. Ex. 1034, 76:19–21.

Even if we were to conclude that the PageRank algorithm utilized the non-semantical method of claim 26, we could not determine whether the alleged success of PageRank is due to its non-semantical aspects, its semantical aspects, or some combination of both.

Patent Owner also points to Google's license of the '352 patent as evidence of nexus. PO Resp. 59–60. Patent Owner, however, admits that this license resulted in the settlement of a lawsuit (*id.*), which without additional contextual evidence, weighs against finding a nexus.

Additionally, we determine that in light of the weak showing of secondary considerations, the evidence of obviousness with respect to Fox Thesis, Fox SMART, and Fox Collection, is sufficient to support the conclusion that claim 26 would have been obvious. *See Leapfrog Enterprises, Inc. v. Fisher-Price, Inc.*, 485 F.3d 1157, 1162 (Fed. Cir. 2007). As discussed above, Petitioner has provided a strong case of obviousness.

Accordingly, even after considering the counter-arguments in Patent Owner's Response, and the evidence cited therein, we find that Petitioner has shown by a preponderance of the evidence that claim 26 is unpatentable as it would have been obvious over the combination of Fox Thesis, Fox SMART, and Fox Collection.

5. Dependent Claims

Petitioner's claim chart persuasively reads all elements of dependent claims 28, 29, 30, 32, 34, and 39 onto the teachings of Fox Thesis, Fox SMART, and Fox Collection, taken together. Pet. 16–23 (citing Ex. 1007, 12, 14–15, 43, 48–49; Ex. 1008, 14, 16, 24–25, 29, 30–33, 36–38, 41, 43–52; Ex. 1009, 1, 15, 17, 23, 126, 151, 173–74, 177–79, 181–82, 191–92, 194, 202, 213–18, 224, 232, 234, 238–43, 257; Ex. 1016 ¶¶ 71–108, 110–

121, 124–125, 127–128, 132–135, 137–145). For instance, we determine that Petitioner has shown by a preponderance of the evidence that the combination of Fox Thesis, Fox SMART, and Fox Collection teaches that the first and second numerical representations are vectors that are arranged in first and second matrices, as required by claim 28. Fox SMART teaches CITED, which is a set of tuples indicating a pair of documents linked by a direct relationship, as well as the direction of citation. Ex. 1008, 29. These tuples are vectors, as are the components of the *bc* and *cc* submatrices, which represent indirect relationships. *Id*. at 30–32. Furthermore, the objects in the CACM database are bibliographic records which include publication date (Ex. 1007, 14), and therefore are assigned chronological data as required by claim 28.

We also determine that Petitioner has shown by a preponderance of the evidence that the combination of Fox Thesis, Fox SMART, and Fox Collection teaches weighing, wherein some indirect relationships are weighed more heavily than other indirect relationships, as recited in claim 32. Fox Thesis, for example, discloses assigning weights to subvectors such as *bc* and *cc*, which are different types of indirect relationships. Ex. 1009, 257 (Table 8.13). In one weighting scheme disclosed in Fox Thesis, bibliographic coupling indirect relationships (*bc*) are weighted at .009, more heavily than co-citation indirect relationships (*cc*), weighted at 0. *Id*.

Claim 39 requires both pool-similarity searching and pool-importance searching. As noted above, we construed *pool-similarity searching* as "identifying at least one object based on degree of similarity to a selected pool of objects" and *pool-importance searching* as "identifying at least one object based on the importance of the object to a selected pool of objects."

We determine that the Fox Papers have been shown by a preponderance of the evidence to teach these elements. Fox Thesis and Fox SMART disclose a feedback search in which results are presented to a user, ranked according to importance, and then used to construct a new search. Ex. 1008, 24, Fig. 6; Ex. 1009, 151, Fig. 5.1. This teaches pool-importance searching as required by claim 39. Similarly, Fox SMART discloses a search that "perform[s] exact matches as well as general similarity computations" (Ex. 1008, 37–38), which meets the pool-similarity searching limitation of claim 39.

Additionally, for the reasons discussed above with respect to claim 26, we determine that Petitioner has satisfied the requirements for combining the teachings of Fox Thesis, Fox SMART, and Fox Collection.

Patent Owner argues that Petitioner has not proven by a preponderance of the evidence that all elements of the dependent claims are taught or suggested by the Fox Papers. PO Resp. 35–39. Some of these arguments, for example those made with respect to claims 28, 29, 30, and 34, are based on the fact that the databases of the Fox Papers do not include objects because the bibliographic records do not cite to one another. *Id*. Just as we found such arguments unpersuasive with respect to claim 26, we are not persuaded by them here. The Fox Papers suggest the inclusion of full text documents into the databases, and that such a modification could be beneficial.

Patent Owner also argues that claim 28's limitation that the step of searching comprises the steps of matrix searching of the second matrices and examining the chronological data is not met by the Fox Papers. *Id*. at 35–36. According to Patent Owner, the Fox Thesis discloses a "preliminary clustering experiment" in which chronological data is "summarily dismissed

because of poor results." *Id*. We do not consider this reading of the Fox Thesis accurate. The quotation from the reference provided by Patent Owner, "the clustering result does not seem as good as that of the other methods" (Ex. 1009, 217), is partial and misleading. The full sentence reads: "*If the* clustering result does not seem as good as that of other methods *then a likely explanation is that improper coefficients were chosen and used in computing the combined similarity value*." *Id*. (omitted portion emphasized). Not only does Patent Owner omit the qualifier "if," but also the explanation that the result likely is due to improper weighting coefficients. This is far from the "summary dismissal" of chronological data asserted by Patent Owner.

Indeed, as Petitioner notes, other portions of the Fox Papers expressly disclose searching using indirect relationship matrices in combination with chronological data. Ex. 1008, 41 (p-norm queries include date, as well as bibliographically coupled or co-cited articles). We are not persuaded by Patent Owner's arguments regarding claim 28.

Patent Owner also contends that the Fox Papers do not teach or suggest marking subsets of objects in the database, as required by claim 34. PO Resp. 38. Fox SMART discloses "separate indexing of paragraphs or even sentences." Ex. 1008, 25; *id.* at 80 ("vectors could be computed for smaller items than just documents"). According to Patent Owner, however, the markings "must be usable by a computerized search to individually identify a specific subset of an object in a computer database as a search result." PO Resp. 38. This alleged requirement is drawn from the *marking* limitation of claim 26. *Id*. Claim 26, however, only requires that the "marked object . . . be individually identified by a computerized search."

The subsets of claim 34 are marked, but this marking does not transform the subsets into objects as recited in claim 26. Patent Owner's argument that computerized searching of the marked subsets is required by claim 34 lacks merit.

Finally, Patent Owner argues that the Fox Papers do not teach or suggest pool-importance searching, as required by claim 39. PO Resp. 39. Patent Owner correctly notes that "importance is distinct from similarity," and therefore pool-importance searching is different than pool-similarity searching. *Id*. The Petition, Patent Owner argues, only identifies disclosures of pool-similarity searching in the Fox Papers, and, therefore, fails to establish that all elements are taught or suggested by the prior art. *Id*.

We disagree with Patent Owner's argument. In its claim chart, Petitioner set forth distinct disclosures from the Fox Papers to meet the pool-importance searching and pool-similarity searching elements. Pet. 22–23. For instance, Fox SMART teaches using "general similarity computations," (Ex. 1008, 37–38) which Petitioner contends is pool-similarity searching, as well as a "feedback" search loop in which results are ranked according to importance to a user, and then further results are retrieved (*id*. at 24, Fig. 6), which Petitioner contends is pool-importance searching. As we concluded above, the feedback search function disclosed in the Fox Papers teaches pool-importance searching, as required by claim 39.

For the foregoing reasons, Petitioner has shown by a preponderance of the evidence that claims 28, 29, 30, 32, 34, and 39 of the '352 patent are unpatentable under 35 U.S.C. § 103(a) as they would have been obvious over Fox Thesis, Fox SMART, and Fox Collection.

D. Obviousness of Claims 26, 28–30, 32, 34, and 39 Over the Tapper Papers

We instituted trial to determine whether claims 26, 28–30, 32, 34, and 39 are unpatentable under 35 U.S.C. § 103 as having been obvious over the combined disclosures of Tapper 1976 and Tapper 1982 (collectively, "the Tapper Papers"). Inst. Dec. 21–24.

We have considered Petitioner's arguments and evidence, as well as the counter-arguments in Patent Owner's Response, and the evidence cited therein, and conclude that Petitioner has not shown by a preponderance of the evidence that each of claims 26, 28–30, 32, 34, and 39 of the '352 patent are unpatentable, under 35 U.S.C. § 103, as having been obvious over the Tapper Papers.

1. Tapper 1976

Tapper 1976 discloses a "citation vector technique" for retrieving legal information that seeks to overcome perceived deficiencies in Boolean search strings. Ex. 1005, 270–71. Rather than characterizing a legal document by the words it contains, vector matching focuses on the citations the document contains. *Id*. at 263. Tapper 1976 also notes that the technique may be used as an adjunct to a full-text retrieval system. *Id*. at 272.

By repeating the vector characterization of the documents, Tapper 1976 discloses that a matrix may be created that shows the similarities between the documents. *Id*. By re-ordering the matrix, the documents may be clustered according to their similarity. *Id*. The reference also discloses that "second generation citations" may be used: "if a case cites cases A', B' and C', and case A' cites a1', a2' and a3', case B' b1', b2' and b3' and case C'

c1', c2' and c3' the original case would be represented by a combination of its own vector, and those of cases A', B' and C'." *Id*. at 266.

2. Tapper 1982

Tapper 1982 similarly focuses on the drawbacks of full-text searching of legal documents and the alternative use of citation vectors for legal research. Ex. 1006, 135–36. The reference discusses weighting certain citation vectors more heavily than others, for example by the difference in the ages of the citing and cited case. *Id*. at 138.

A pilot project implementing such a citation vector-based system is also described by Tapper 1982. *Id*. at 139. The reference discloses a correlation algorithm used in the pilot project to cluster together vectors with a high degree of association. *Id*. at 143–44. Such clustering is said to permit a document to be retrieved "not only because it is itself closely associated with another target document, but also because both it and the target document are closely associated with a third." *Id*.

3. Claim 26

As discussed above, claim 26 requires steps of "creating a first numerical representation for each identified object in the database based upon the object's direct relationship with other objects in the database." We find that this limitation is neither taught nor suggested by the combined Tapper Papers.

Petitioner's claim chart identifies several portions which allegedly teach a first numerical representation. Pet. 47–48. For example, Tapper 1976 is cited as disclosing "quantifiable representation in the form of numerical weighting" *Id*. (citing Ex. 1005, 263). These "quantifiable representations" are of "other characteristics" of the citations, such as age or

the importance of the court or jurisdiction deciding the case, not the citations (relationships) themselves. Ex. 1005, 263.

Similarly, Tapper 1982 is cited as disclosing "[a]scription of numerical values to vector elements." Pet. 48 (citing Ex. 1006, 141). But Tapper 1982 explicitly defines "vectors" as "the strings [a document] contains and the frequency of their occurrence." Ex. 1006, 134. In other words, the "numerical values" of Tapper 1982 are the *frequency* of the appearance of citation *strings*, which as we discussed above, connotes the inclusion of letters. The Petition provides no citation to either Tapper Paper that teaches representing direct relationships with a first numerical representation.

In its Reply Brief, Petitioner identifies two other disclosures by the Tapper Paper it contends satisfy the *first numerical representation* limitation. First, Petitioner argues that "the legal citations in Tapper clearly qualify as numerical representations." Reply 4–5. The legal citations Petitioner identifies, however, are in the exemplary form of "500 F.2d 411," which includes letters. As we have construed the term, this is not a numerical representation, but rather the "strings" of the vectors discussed above.

Second, Petitioner notes that the Tapper Papers describe assigning cases in the database a unique ID number. *Id*. (citing Ex. 1006, 148). At oral argument, Petitioner's counsel directed our attention to Table 2 of Tapper 1982, which includes in the leftmost column pairs of numbers which signify pairs of documents. Tr. 14; Ex. 1006, 147. At most, the assignment of these numbers could satisfy the marking step of claim 26 (Ex. 1006, 148 ("[t]he first column gives the numbers allocated to the cases.")); they are not

generating a first numerical representation. The document numbers indicated by Petitioner are numerical representations of *documents*, not of the *relationships* between those documents. Claim 26 requires that the first numerical representations are based on direct relationships in the database. The numbers allocated to the cases of Tapper 1982 cannot satisfy this limitation.

Nor can the document number *pairs* of Table 2 be a first numerical representation, as Tapper 1982 does not disclose that they represent a direct relationship (i.e., one of the documents in the pair citing the second). Rather, the pairs of documents appear to be listed together in the table because of their high "correlation values." Ex. 1006, 148. As Petitioner acknowledges, these correlation values represent indirect relationships between the documents (Reply 6 ("correlation values of cases' indirect relationships")), therefore they cannot be a first numerical representation that represents a direct relationship.

Petitioner argues in the alternative that "there is nothing non-obvious about creating citation vectors consisting solely of numbers." Reply 4–5. At the outset, we note that this argument was presented for the first time in the Reply; the sole modification to the Tapper Papers addressed in the Petition is the combination of the disclosures of the two references. Pet. 45–46. Nor did Petitioner present any testimony with the Petition regarding the Tapper Papers, or how a person of ordinary skill in the art would have modified the references. It would be a proper exercise of our discretion, therefore, to not consider this argument and the Reply Declaration of Dr. Fox (Ex. 1016), which presents testimony on the Tapper Papers for the first

time.[3] *See* Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (Aug. 14, 2012) ("a reply that raises a new issue or belatedly presents evidence will not be considered.")

Even if we were to consider Petitioner's Reply and Dr. Fox's Reply Declaration, however, we are not persuaded. Petitioner cites to various portions of the Tapper Papers (Reply 4–5), but none of these citations sufficiently establish a reason to substitute numerical representations for those disclosed in Tapper. For example, Petitioner argues—using pieced-together quotations—that "Tapper [1982] also makes clear that one could 'very easily' use a 'simple conversion table' to map 'extracted' citations to any 'chosen style.'" *Id*. at 5 (citing Ex. 1006, 136). Upon reading the full context from which these quotes are drawn, however, it is clear that Tapper 1982 is discussing "parallel reports of the same decision." Ex. 1006, 136. In other words, Tapper 1982 does not contemplate converting letter-containing case citations into numbers, but rather converting one letter-containing citation into another.

Dr. Fox's Reply Declaration (Ex. 1030 ¶¶ 107–115) relies on the same arguments as Petitioner's Reply, and we find them unpersuasive for the same reasons. Nor are we persuaded by the portions of Dr. Jacobs's cross-examination Petitioner cites (Reply 5 (citing Ex. 1033, 313:7–316:23, 339:3–342:6)), as Dr. Jacobs's testimony was to what a person of ordinary skill would have understood from the '352 patent specification, not the Tapper Papers. *See In re Vaeck*, 947 F.2d 488, 493 (Fed. Cir. 1991) (suggestion to make invention cannot "be founded . . . in the applicant's

[3] We address Patent Owner's Motion to Exclude portions of the Reply Declaration below.

disclosure"). The record before us does not support the conclusion that a person of ordinary skill in the art would have modified the combined disclosures of the Tapper Papers to include a first numerical representation.

4. Dependent Claims

The remaining instituted claims all depend, directly or indirectly, from claim 26, and thus incorporate claim 26's requirement of a first numerical representation. We, therefore, find that the Tapper Papers do not teach or suggest all elements of these dependent claims.

E. Motion to Exclude

Patent Owner filed a Motion to Exclude (Paper 47) in which Patent Owner seeks to exclude portions of the Reply Declaration of Dr. Edward A. Fox (Ex. 1030) ("Reply Declaration") submitted with Petitioner's Reply. In particular, Patent Owner identifies three issues with the Declaration, each of which is based on the argument that portions of the Declaration are improper reply evidence.

In its Reply, a Petitioner may only respond to arguments raised in the Patent Owner's Response. 37 C.F.R. § 42.23(a). "A reply that raises a new issue or belatedly presents evidence will not be considered." Office Patent Trial Practice Guide, 77 Fed. Reg. at 48,767 (Aug. 14, 2012). The Practice Guide provides, as indications of improper reply evidence, "new evidence necessary to make out a *prima facie* case for . . . patentability or unpatentability . . ., and new evidence that could have been presented in a prior filing." *Id.*

A motion to exclude evidence under 37 C.F.R. § 42.64(c), however, "normally is not the proper vehicle for resolution of a dispute regarding reply arguments and evidence exceeding the proper scope of a reply." *ABB,*

Inc. v. Roy-G-Biv Corp., Case IPR2013-00063, slip op. 13–14 (PTAB May 16, 2014) (Paper 71); *Corning Inc. v. DSM IP Assets B.V.*, Case IPR2013-00047, slip op 7 n.3 (PTAB May 1, 2014) (Paper 84) (characterizing such motions as "now disfavored"). Rather, when evaluating the record after oral argument, the Board is capable of determining what, if any, evidence exceeds the proper scope of rely, and accordingly disregarding that evidence.

While we, therefore, *deny* Patent Owner's Motion, we also note that even if it were proper, we would dismiss it as moot. With respect to the objected-to portions of the Reply Declaration which discuss the Tapper Papers, we have considered them above, found Dr. Fox's testimony unpersuasive, and found in favor of Patent Owner on the Tapper Papers ground. With respect to the Fox Papers ground, we have found in favor of Petitioner, but did not rely on any of the objected-to portions of the Reply Declaration in so doing. A decision to exclude the Reply Declaration would, therefore, not affect our determinations in this case.

F. Motions to Seal

Patent Owner filed a Motion to Seal (Paper 35) the Declaration of Dr. Amy N. Langville ("Langville Declaration") filed as Exhibit 2114. Petitioner filed a Motion to Seal (Paper 42) the Transcript of the Deposition of Amy N. Langville, Ph.D. ("Langville Transcript") filed as Exhibit 1034. Both of these motions are unopposed.

Regarding Patent Owner's Motion to Seal, according to Patent Owner paragraphs 25, 112, and 113 of the Langville Declaration makes reference to certain facts about confidential licenses to the patents under review. Paper

35, 3. Additionally, Patent Owner contends that this information has not been made, and will not be made, public. *Id*.

Regarding Petitioner's Motion to Seal, according to Petitioner, Patent Owner has designated the transcript as confidential. Paper 42, 3. To avoid public disclosure, therefore, Petitioner submits sealing the Langville Transcript is appropriate. *Id*.

There is a strong public policy in favor of making information filed in *inter partes* review proceedings open to the public. *See Garmin Int'l v. Cuozzo Speed Techs., LLC*, Case IPR2012-00001 (PTAB Mar. 14, 2013) (Paper 34). Under 35 U.S.C. § 316(a)(1), the default rule is that all papers filed in an *inter partes* review are open and available for access by the public.[4] The standard for granting a motion to seal is "good cause." 37 C.F.R. § 42.54. A moving party bears the burden of showing that the relief requested should be granted. 37 C.F.R. § 42.20(c).

Regarding Patent Owner's Motion to Seal, Patent Owner, as the moving party, has failed to carry its burden. Patent Owner identifies only three paragraphs in the Langville Declaration that purportedly contain confidential information. However, Patent Owner has not pointed to proof in the record that any information contained in these paragraphs is confidential. Additionally, although Patent Owner contends that this information has not been made, and will not be made, public, Patent Owner presented this information during the hearing on October 30, 2014, which

---

[4] Additionally, we note that confidential information subject to a protective order ordinarily would become public 45 days after final judgment in a trial. Office Patent Trial Practice Guide, 77 Fed. Reg. at 48,761. However, after denial of a petition to institute a trial or after final judgment in a trial, a party may file a motion to expunge confidential information from the record. 37 C.F.R. § 42.56.

was open to the public. *See* Tr. 54:12–25. We, therefore, determine that Patent Owner has not met its burden of proof.

Regarding Petitioner's Motion to Seal, Patent Owner's designation of the transcript as confidential is not sufficient to show that the transcript contains confidential information. We, therefore, determine that Petitioner has not met its burden of proof.

We recognize a denial of the motions to seal would unseal immediately the material that Patent Owner desires to remain confidential and the effect would be irreversible. Therefore, rather than denying the motions at this time, we will provide Patent Owner and Petitioner one week to (1) withdraw the motions to seal and request that we expunge Exhibits 2114 and 1034, or (2) withdraw the motions to seal, request that we expunge Exhibits 2114 and 1034, and replace them with redacted versions that leave out the confidential information. We note that we have not relied on the three paragraphs of the Langville Declaration that Patent Owner identifies as containing allegedly confidential information.

III. CONCLUSION

We conclude that Petitioner has shown by a preponderance of the evidence that claims 26, 28–30, 32, 34, and 39 of the '352 patent are unpatentable under 35 U.S.C. § 103, as they would have been obvious over Fox Thesis, Fox SMART, and Fox Collection, taken together.

IV. ORDER

For the reasons given, it is

ORDERED that claims 26, 28–30, 32, 34, and 39 of U.S. Patent No. 5,544,352 are determined by a preponderance of the evidence to be unpatentable;

FURTHER ORDERED that Patent Owner's Motion to Exclude the Reply Declaration of Dr. Edward A. Fox (Exhibit 1030) is denied;

FURTHER ORDERED that Exhibit 2114 and Exhibit 1034 will be made available to the public after 5 PM Eastern five business days after the entry date of this decision, unless prior to that time, each of Patent Owner and Petitioner (1) withdraws the motions to seal and requests that we expunge Exhibits 2114 and 1034, or (2) withdraws the motions to seal, requests that we expunge Exhibits 2114 and 1034, and replaces them with redacted versions that leave out the confidential information; and

FURTHER ORDERED that, because this is a final written decision, parties to the proceeding seeking judicial review of the decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

FOR PETITIONER:

Heidi L. Keefe
COOLEY, LLP
hkeefe@cooley.com
dcpatentdocketing@cooley.com.

David Silbert
KEKER & VAN NEST, LLP
djs@kvn.com
efiling@kvn.com

FOR PATENT OWNER:

Martin M. Zoltick
Nancy J. Linck
Soumya P. Panda
ROTHWELL, FIGG, ERNST & MANBECK, P.C.
mzoltick@rfem.com
nlinck@rfem.com
SRA-IPR@rfem.com